WILLIAM L. ANTHONY (State Bar No. 106908)
ERIC L. WESENBERG (State Bar No. 139696)
HEIDI L. KEEFE (State Bar No. 178960)
ORRICK, HERRINGTON & SUTCLIFFE, LLP
1000 Marsh Road
Menlo Park, CA 94025
Telephone: 650-614-7400
Facsimile: 650-614-7401

STEVEN ALEXANDER (admitted *Pro Hac Vice*)
KRISTIN L. CLEVELAND (admitted *Pro Hac Vice*)
JAMES E. GERINGER (admitted *Pro Hac Vice*)
JOHN D. VANDENBERG
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, OR 97204
Telephone: 503-226-7391
Facsimile: 503-228-9446

Attorneys for Defendant and Counterclaimant,
MICROSOFT CORPORATION

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

OAKLAND DIVISION

INTERTRUST TECHNOLOGIES CORPORATION, a Delaware corporation, Plaintiff, v. MICROSOFT CORPORATION, a Washington corporation, Defendant.	CASE NO. C01-1640 SBA **MICROSOFT CORPORATION'S PATENT LOCAL RULE 4-2 DISCLOSURE OF PRELIMINARY CLAIM CONSTRUCTION AND EXTRINSIC EVIDENCE (LIMITED TO "MINI-MARKMAN" CLAIMS)**
MICROSOFT CORPORATION, a Washington corporation, Counterclaimant, v. INTERTRUST TECHNOLOGIES CORPORATION, a Delaware corporation, Counter Claim-Defendant.	

Pursuant to Patent Local Rule 4-2 and this Court's Order, entered November 5, 2002, Defendant Microsoft Corporation ("Microsoft") hereby serves its "Disclosure Of Preliminary Claim Construction And Extrinsic Evidence," limited to the twelve selected "Mini-Markman" patent claims. Microsoft's preliminary claim construction is based upon the proposed terms, phrases and clauses, and claims as a whole, identified by the parties in their submissions in accordance with Patent Local Rule 4-1(a) and conference in accordance with Patent Local Rule 4-1(b).

Microsoft provides its preliminary claim construction of each of the 12 "Mini-Markman" claims subject to the limitations and reservations of rights set forth herein. Microsoft does not waive any defenses that the asserted claims fail to satisfy the provisions of 35 U.S.C. § 112 including, for example, the written description requirement, the definiteness requirement, or any other requirement for patentability. Microsoft does not concede that the asserted claims are supported by Plaintiff's original application or any application from which they purportedly claim priority. Specifically, by offering a construction of a term, Microsoft does not waive any defense that the claim is in fact indefinite and there can be no proper construction.

Microsoft provides its preliminary claim construction in the following format. Exhibit A sets forth Microsoft's preliminary construction of (1) the claim term "virtual distribution environment" ("VDE"), (2) the "VDE invention" disclosed in the February, 1995, InterTrust patent application, and (3) certain other claim terms. Exhibit B sets forth Microsoft's preliminary construction of the disputed claims as a whole, and particular claim phrases in dispute, in the order of appearance in a claim. Where an individual claim term (within a phrase) is also in dispute, it will be bold-faced in Exhibits A and B. Exhibit C sets forth Microsoft's preliminary construction of the individual terms in dispute, in alphabetical order.

Microsoft reserves the right to modify its preliminary claim constructions in the event that the parties are unable to agree upon a particular claim construction. Furthermore, because InterTrust has not yet fully complied with the disclosure requirements of Patent Local Rules 3-1 and 3-2, Microsoft expressly reserves the right to amend its preliminary claim construction if

evidence becomes available through those disclosures (or that should have been provided therein) that would support amended constructions. Microsoft further reserves the right to amend its preliminary claim constructions once it has an opportunity to review InterTrust's preliminary claim constructions and once the parties have further met and conferred as required.

Preliminary Identification of Evidence in Support of Claim Construction

Microsoft's preliminary claim construction is supported by the intrinsic record of the seven U.S. patents from which the 12 "Mini-Markman" claims are selected. For the purposes of submission of this preliminary claim construction only, Microsoft treats the "intrinsic" evidence as including: 1) the specifications of each of the seven U.S. patents at issue in the "Mini-Markman" proceeding, including any material purportedly incorporated by reference therein; 2) the prosecution history of each of the seven patents at issue, including the applications and prosecution history of the seven patents and any related patent applications, including without limitation, applications purportedly incorporated by reference or to which an application claimed priority; and 3) all references cited in the prosecution of any such applications. In accordance with the local rules, this evidence is not specifically identified, except to the extent that Microsoft asserts particular sections of a patents' specifications provide "structure" for claims properly construed under 35 U.S.C. § 112(6).

In certain circumstances, Microsoft's preliminary construction may be supported by extrinsic evidence presently available to Microsoft. Microsoft reserves the right to modify or supplement with evidence that it has not yet been able to fully review, due to InterTrust's production, including without limitation, InterTrust re-production of over 1,000,000 pages on November 4, 2002. Microsoft reserves the right to supplement with additional evidence gathered in the course of the discovery collected between now and the close of "claim construction" discovery or later submitted by InterTrust in full compliance with its disclosure obligations under Patent Local Rules 3-1 and 3-2. Extrinsic evidence is identified or produced in accordance with the local rule and set forth in the following exhibits:

Exhibit D: Contains copies of excerpts from dictionaries and other publications. Due to the volume of the appended pages, Exhibit D will be served via Federal Express.

Exhibit E: Contains a list of selected production documents, identified by initial bates number.

Exhibit F: Contains a list of selected, uncited prior art publications, identified by bates number(s).

Exhbit G: Contains a list of selected, uncited prior art patents, identified by bates number(s).

In addition to the extrinsic evidence cited in Exhibits D-G, Microsoft incorporates by reference herein and reserves the right to rely upon: 1) all documents identified by InterTrust in response to discovery or pursuant to the Patent Local Rules; 2) all InterTrust patents, publications and other things that are prior art to any Mini-Markman claim; and 3) the testimony of InterTrust and the witnesses identified below.

Preliminary Identification of Witnesses[1]

Professor John Mitchell: Dr. Mitchell will testify of the following matters:

1) that certain of the presently disputed terms and phrases used in the twelve claims are amorphous terms lacking a well-defined, precise meaning that can accurately be gleaned from technical or other dictionaries. Rather, these terms are used in the art and/or in the patents in a manner that requires close consideration of the entire patent specification to put them in proper context and determine their precise, correct meaning as used in the patents. These terms include "secure container," "control," "govern," "protect," "protected processing environment," "secure," "securely," "security," "virtual distribution environment";

2) that the concepts stated in the InterTrust patents were known to the art, including the cited prior art, which cited art he will describe;

[1] In accordance with the local rules, Microsoft identifies witness testimony that it contends will support its construction. It has not identified herein testimony relevant to the "tutorial" to be held prior to the claim construction hearing.

3) the level of skill, background, and understanding (including extent thereof) of the relevant patent application disclosures by a person of skill in the art; and

4) the meaning and scope certain disputed claim language, including "secure container," "control," "govern," "protect," "protected processing environment," "secure," "securely," "security," and "virtual distribution environment."

Professor David Maier: Dr. Maier will testify on the following matters:

1) what the February 13, 1995, patent application (SN 08/388,107) and the seven InterTrust patents, described as the "invention;" more particularly, what are the required, necessary, non-optional features of the "VDE" "invention" as stated in the patents. This description will include an explanation of the features set forth in Microsoft's "Global Constructions" (Exhibit A).

2) what the February 13, 1995, patent application (SN 08/388,107) and the seven InterTrust patents, required as necessary, non-optional building blocks to implement the "VDE" "invention" as stated in the patents.

Dated: December 20, 2002

By: 

WILLIAM L. ANTHONY
ERIC L. WESENBERG
HEIDI L. KEEFE
ORRICK HERRINGTON & SUTCLIFFE, LLP
1000 Marsh Road
Menlo Park, CA 94025
Telephone: 650-614-7400

STEVEN ALEXANDER
KRISTIN L. CLEVELAND
JAMES E. GERINGER
JOHN D. VANDENBERG
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, OR 97204
Telephone: 503-226-7391

Attorneys for Defendant
MICROSOFT CORPORATION

Of Counsel:

T. Andrew Culbert, Esq.
Microsoft Corporation
One Microsoft Way
Building 8
Redmond, WA 98052-6399
Telephone: 425-882-8080

DECLARATION OF SERVICE BY E-MAIL

I am more than eighteen years old and not a party to this action. My place of employment and business address is 121 S.W. Salmon St., Suite 1600, Portland, OR 97204.

On December 20, 2002, at 3:00 p.m., I served on counsel for InterTrust Technologies Corporation:

MICROSOFT CORPORATION'S PATENT LOCAL RULE 4-2 DISCLOSURE OF PRELIMINARY CLAIM CONSTRUCTION AND EXTRINSIC EVIDENCE (LIMITED TO "MINI-MARKMAN" CLAIMS)

by email delivery to:

Michael H. Page, Esq. John W. Keker, Esq. Jon B. Streeter, Esq. Keker & Van Nest, LLP 710 Sansome Street San Francisco, CA 94111 Telephone: 415-391-5400 Facsimile: 415-397-7188 Email: mhp@kvn.com	Douglas Derwin InterTrust Technologies Corporation 4800 Patrick Henry Drive Santa Clara, CA 95054 Telephone: 408-855-0100 Facsimile: 408-855-0144 Email: dderwin@intertrust.com
Steven H. Morrissett, Esq. Finnegan Henderson Farabow Garrett & Dunner Stanford Research Park 700 Hansen Way Palo Alto CA 94304-1016 Telephone: 202-408-4000 Facsimile: 202-408-4400 Email: steven.morrissett@finnegan.com	

I declare under penalty of perjury that the foregoing is true and correct.

Executed on December 20, 2002, at Portland, Oregon.

(SIGNATURE)

Kristin L. Cleveland

(PRINT NAME)

Microsoft PLR 4-2, Exhibit A: Mini Markman Preliminary Claim Construction -- Global

Below is Microsoft's preliminary construction of the "Virtual Distribution Environment" ("VDE") "invention" of the February 13, 1995, InterTrust application (the "VDE Invention") and certain other terms, which constructions are incorporated by reference into Microsoft's preliminary construction of certain other disputed claims, claim terms, and claim phrases.[1]

Required Feature	Construction
Security and Commerce World	InterTrust's February 13, 1995, patent application described as its "invention" a Virtual Distribution Environment ("VDE invention") for **securing**, administering, and **auditing** all **security** and commerce digital information within its multi-node world (community). VDE guarantees to all VDE "participants" identified in the patent application that it will maintain the **availability**, **secrecy**, **integrity** and **authenticity** of all such information present at any appliance (node) within the VDE world (including protected content (including currency, credit, payments, etc.), information about content usage, content-**control information**, **controls**, **load modules**, etc.). VDE is **secure** against at least the threats identified in the patent application to this **availability** (no user may delete the information without **authorization**), **secrecy** (neither available nor disclosed to unauthorized persons or processes), **integrity** (neither intentional nor accidental alteration), and **authenticity** (asserted characteristics are genuine). VDE further provides and requires the components and capabilities described below. Anything less than or different than this is not VDE or the described "invention."
VDE Secure Processing Environment	At each node where VDE-protected information is **accessed**, **used**, or assigned **control information**, VDE requires a Secure Processing Environment. A Secure Processing Environment is uniquely identifiable, self-contained, non-circumventable, and trusted by all other VDE nodes to protect the **availability**, **secrecy**, **integrity** and **authenticity** of all information identified in the patent application as being protected, and to guarantee that such information will be **accessed** and **used** only as expressly **authorized** by the **associated VDE controls**. A Secure Processing Environment is formed by, and requires, a special-purpose **Secure Processing Unit** having a hardware **tamper-resistant barrier encapsulating** a processor and internal **secure memory**. The **barrier** prevents all **unauthorized** interference, removal, observation, and **use** of the information and processes within it. A Secure Processing Environment is under **control of controls** and **control information** provided by one or more parties, rather than being under **control** of the appliance's users or programs.
VDE Controls	VDE **allows access** to or **use** of protected information and processes only through execution of (and satisfaction of the requirements imposed by) independent,

[1] The word "invention" is used not to suggest that anything described in InterTrust's patents in fact was novel or non-obvious or inventive, but rather to identify what was described as the **alleged** invention. Also, features and capabilities are described as they are described in the InterTrust patent application, even though the patent application did not describe an actual working system having any of these capabilities. Also, Microsoft's proposed constructions use many terms from the InterTrust patents that are used inconsistently or otherwise indefinitely in the patents. Those terms are used by Microsoft in their narrowest applicable sense, and without waiving the right to assert the indefiniteness of this claim language. Also, the preliminary constructions assume (without conceding) that the February, 1995, InterTrust patent application was incorporated by reference into the '721, '861, and '683 patents, effectively for claim construction purposes. If the Court concludes otherwise, then the proper constructions will be different in some cases. Bolded terms are preliminarily defined in Exhibits A-C of Microsoft's PLR 4-2 papers.

Required Feature	Construction
	special-purpose, executable VDE control(s). A VDE control can execute only within a **Secure Processing Environment**. Each VDE control is a **component assembly** dedicated to a particular activity (e.g., editing, modifying another control, a user-defined action, etc.), particular user(s), and particular protected information. Each separate information **access** or **use** is independently **controlled** by independent VDE control(s). Each VDE control is assembled, within a **Secure Processing Environment**, from independently deliverable modular components (e.g., **load modules** or other controls), dynamically in response to an information **access** or **use request**. The dynamic assembly of a control is directed by a "blueprint" **record** (put in place by one or more VDE users) **containing control information** identifying the exact modular code components to be assembled and executed to **govern** this particular activity on this particular information by this particular user(s). Each control is independently assembled, loaded and delivered vis-à-vis other controls. **Control information** and controls are extensible and can be configured and modified by all users, and combined by all users with any other VDE **control information** or controls (including that provided by other users), subject only to "senior" user controls. Users can assign **control information** and controls to an arbitrarily fine, user-defined portion of the protected information, such as a single paragraph of a document, as opposed to being limited to file-based controls. VDE controls reliably limit use of the protected information to **authorized** activities and amounts.
VDE Secure Containers	A VDE secure container is a self-contained, self-protecting data structure which (a) encapsulates information of arbitrary size, type, format, and organization, including other, nested, containers, (b) cryptographically protects that information from all **unauthorized access** and **use**, (c) provides encrypted storage management functions for that information, such as hiding the physical storage location(s) of its protected contents, (d) permits the **association** of itself or its contents with **controls** and **control information governing access** to and **use** thereof, and (e) **prevents** such **use** or **access** (as opposed to merely **preventing** decryption) until it is **opened**. A secure container can be **opened** only as expressly **allowed** by the **associated VDE control(s)**, only within a **Secure Processing Environment**, and only through decryption of its encrypted header. A secure container is not directly accessible to any non-VDE calling process. All such calls are intercepted by VDE. The creator of a secure container can assign (or allow others to assign) **control information** to any arbitrary portion of a secure container's contents, or to an empty secure container (to **govern** the addition of contents to the secure container, and **access** to or **use** of those contents). A container is not a secure container merely because its contents are encrypted and signed. A secure container is itself **secure**. All VDE-protected information (including protected content, information about content usage, content-**control information, controls**, and **load modules**) is encapsulated within a secure container whenever stored outside a **Secure Processing Environment** or **secure database**.
Non-Circumventable	VDE is non-circumventable (sequestered). It intercepts all attempts by any and all users, processes, and devices, to **access** or **use** (e.g., observe, interfere with, or remove) protected information, and prevents all such attempts other than as allowed by execution of (and satisfaction of all requirements imposed by) **associated VDE controls** within **Secure Processing Environment(s)**.
Peer to Peer	VDE is peer-to-peer. Each VDE node has the innate ability to perform any role identified in the patent application (e.g., end user, content packager, distributor, **clearinghouse**, etc.), and can protect information flowing in any direction between any nodes. VDE is not client-server. It does not pre-designate and restrict one or more nodes to act solely as a "server" (a provider of information (e.g., authored content, **control information**, etc.) to other nodes) or "client" (a requestor of such information). All types of protected-content transactions can proceed without requiring interaction with any server.
Comprehensive Range of	VDE comprehensively governs all **security** and commerce activities identified in the patent application, including (a) metering, budgeting, monitoring, reporting, and **auditing** information usage, (b) billing and paying for information usage, and (c) negotiating, signing and enforcing contracts that establish users' rights to

Required Feature	Construction
Functions	access or use information.
User-Configurable	The specific protections governing specific VDE-protected information are specified, modified, and negotiated by VDE's users . For example, VDE enables a consumer to place limits on the nature of content that may be accessed at her node (e.g., no R-rated material) or the amount of money she can spend on viewing certain content, both subject only to other users' senior controls.
General Purpose; Universal	VDE is universal as opposed to being limited to or requiring any particular type of appliance, information, or commerce model. It is a single, unified standard and environment within which an unlimited range of electronic rights protection, data security, electronic currency, and banking applications can run.
Flexible	VDE is more flexible than traditional information security and commerce systems. For example, VDE allows consumers to pay for only the user-defined portion of information that the user actually uses, and to pay only in proportion to any quantifiable VDE event (e.g., for only the number of paragraphs displayed from a book).

Microsoft PLR 4-2, Exhibit B: Mini Markman Preliminary Claim Construction – Claim Terms[1]

Claim Term	Preliminary Construction
access, accessed, access to, accessing	Establishing the connections, routings, and **security** requisites needed to physically obtain something. Access to protected information is required, but insufficient, for use of that information. In VDE, access to protected information is achieved only through execution (within a **Secure Processing Environment**) of the **VDE control**(s) assigned to the particular "access" request, satisfaction of all requirements imposed by such execution, and the **controlled opening** of the **secure container containing** the information.
addressing	Referring by specific location or individual name to something without physically storing it.
allowing, allows	Actively permitting an action that otherwise cannot be taken (i.e., is absolutely prohibited) by any user, process, or device. In VDE, an action is allowed only through execution (within a **Secure Processing Environment**) of the **VDE control**(s) assigned to the particular action request, and satisfaction of all requirements imposed by such execution.
applying . . . in combination	[This shall be construed in connection with a disputed claim phrase.]
arrangement	[This shall be construed in connection with a disputed claim phrase.]
aspect	[This shall be construed in connection with a disputed claim phrase.]
associated with	1. A specific, direct, persistent, and binding relationship with one or more discrete items. Code that processes information but is merely a general-purpose component of an installation is not "associated with" that information. In VDE, an association between a unit of **executable** code and particular information, or between particular **control information** and a **secure container**, cannot be broken except as **allowed** by execution (within a **Secure Processing Environment**) of assigned **VDE control(s)** and satisfaction of all requirements imposed by such execution. 2. Associations in **VDE** are created with a **component assembly**, a **secure container**, a **Secure Processing Environment**, "object registration," and other mechanisms of **VDE** for (allegedly) individually ensuring the "access control" "handcuffs" between specific **controls**, specific objects (and their content at an arbitrary granular level), and specific users.

[1] The word "invention" is used not to suggest that anything described in InterTrust's patents in fact was novel or non-obvious or inventive, but rather to identify what was described as the **alleged** invention. Also, features and capabilities are described as they are described in the InterTrust patent application, even though the patent application did not describe an actual working system having any of these capabilities. Also, Microsoft's proposed constructions use many terms from the InterTrust patents that are used inconsistently or otherwise indefinitely in the patents. Those terms are used by Microsoft in their narrowest applicable sense, and without waiving the right to assert the indefiniteness of this claim language. Also, the preliminary constructions assume (without conceding) that the February, 1995, InterTrust patent application was incorporated by reference into the '721, '861, and '683 patents, effectively for claim construction purposes. If the Court concludes otherwise, then the proper constructions will be different in some cases. Bolded terms are preliminarily defined in Exhibits A-C of Microsoft's PLR 4-2 papers.

Claim Term	Preliminary Construction
authentication	The act of verifying credentials designed to vouch for the **authenticity** of the identity, data **integrity**, and origin **integrity** of a person, device, program, information, or process.
authorization information, authorized, not authorized	authorized: An action is permitted that otherwise cannot be taken (i.e., is absolutely prohibited) by any user, process, or device. In VDE, an action is authorized only through execution of the applicable **VDE control**(s) within a **VDE Secure Processing Environment** and satisfaction of all requirements imposed by such execution. authorization information: **"Control information"** identifying the exact modular code components to be assembled into a **VDE control** and executed within a **Secure Processing Environment** to permit a particular activity that otherwise cannot be taken (i.e., is absolutely prohibited). ("Control information" is information which identifies the exact modular code components and data which must be assembled and executed to **control** a particular activity on particular information, of arbitrary, user-defined granularity, by particular user(s)). "not authorized": The action is prohibited and cannot be taken by any user, process, or device.
budget control; budget	budget: A unique type of **"method"** that specifies limitations on future usage (e.g., copying) of digital information and how such usage will be paid for, if at all. (A "method" is a collection of basic instructions, and information related to basic instructions, that provides context, data, requirements, and/or relationships for use in performing, and/or preparing to perform, basic instructions in relation to the operation of one or more electronic appliances.) budget control: A **VDE control** assembled using a budget, and enforcing that budget. No process, user, or device is able to make the use identified by the budget once the budget's specified limitation on that **use** has been met.
can be	Something is permitted that otherwise cannot happen (i.e., is absolutely prohibited).
capacity	Available storage space that is still capable of allocation. For example, a 650 MB blank CD, after sealing, has zero capacity because no new material may be stored within it.
clearinghouse	A computer system that provides intermediate storing and forwarding services for both content and **audit information**, and which two or more parties trust to provide its services independently because it is operated under constraint of **VDE security**. "Audit information" means all information created, stored, or reported in connection with an **"auditing"** process. "Auditing" means tracking, metering and reporting the usage of particular information or a particular appliance.
compares, comparison	A processor operation that evaluates two quantities and sets one of three flag conditions as a result of the comparison – greater than, less than, equal to.
component assembly (2)	A cohesive executable component created by a channel which binds or links together two or more independently deliverable **load modules**, and associated data. A component assembly is assembled, and executes only within a **VDE Secure Processing Environment**. A component assembly is assembled dynamically in response to, and to service, a particular content-related activity (e.g., **use** request). Each VDE component assembly is assigned and dedicated to a particular activity, particular user(s), and particular protected information. Each component assembly is independently

Claim Term	Preliminary Construction
	assembled, loadable and deliverable vis-à-vis other component assemblies. The dynamic assembly of a component assembly is directed by a "blueprint" record **containing control information for** this particular activity on this particular information by this particular user(s). Component assemblies are extensible and can be configured and reconfigured (modified) by all users, and combined by all users with other component assemblies, subject only to other users' "senior" **controls.**
contain, contained, containing	Physically storing within, as opposed to **addressing.**
control (n.), controls (n.) (2 – 193:1,11,15,19; 891:1)	VDE **allows access** to or **use** of protected information only through execution of (and satisfaction of the requirements imposed by) independent, special-purpose, executable VDE control(s). A VDE control can execute only within a **Secure Processing Environment**. Each VDE control is a **component assembly** dedicated to a particular activity (e.g., editing, modifying another control, a user-defined action, etc.), particular user(s), and particular protected information. Each separate information **access** or **use** is independently **controlled** by independent VDE control(s). Each VDE control is assembled within a **Secure Processing Environment** from independently deliverable modular components (e.g., **load modules** or other controls), dynamically in response to an information **access** or use **request**. The dynamic assembly of a control is directed by a "blueprint" **record** (put in place by one or more VDE users) **containing control information** identifying the exact modular code components to be assembled and executed to **govern** this particular activity on this particular information by this particular user(s). Each control is independently assembled, loaded and delivered vis-à-vis other controls. **Control information** and controls are extensible and can be configured and modified by all users, and combined by all users with any other VDE **control information** or controls (including that provided by other users), subject only to "senior" user controls. Users can assign **control information** (including alternative **control information**) and **controls** to an arbitrarily fine, user-defined portion of the protected information , such as a single paragraph of a document, as opposed to being limited to file-based controls. VDE controls reliably limit use of the protected information to **authorized** activities and amounts.
controlling, control (v.)	1. Reliably defining and enforcing the conditions and requirements under which an action that otherwise absolutely cannot be taken, will be **allowed,** and the manner in which it may occur. Absent verified satisfaction of those conditions and requirements, the action cannot be taken by any user, process or device. In VDE, an action is controlled through execution of the applicable **VDE control**(s) within a **VDE Secure Processing Environment.** 2. More specifically, in VDE, **controlling** is effected by use of **VDE controls, VDE secure containers,** and VDE foundation (including **VDE Secure Processing Environment,** "object registration," and other mechanisms for allegedly individually ensuring that specific **controls** are enforced vis-à-vis specific objects (and their content at an arbitrary granular level) and specific "users.")
copied file	A **digital file** which has been **copied** at least once, not the **copy** itself. A "copy" is what is formed by a **copying** operation, and it may or may not be encrypted, ephemeral, usable, or accessible.
copy, copied, copying (v.)	To duplicate a **digital file** or other complete physical block of data from one location on a storage medium to another location on the same or different storage medium, leaving the original block of data unchanged, such that two distinct and independent objects exist. Although the layout of the data values in physical storage may differ from the original, the resulting "copy" is logically indistinguishable from the original. The resulting "copy" may or

Claim Term	Preliminary Construction
	may not be encrypted, ephemeral, usable, or accessible.
copy control	A **VDE control** which **controls** some **access** to or **use** of a **copy**.
creating, creation	[This shall be construed in connection with a disputed claim phrase.]
data item	An individual unit of information representing a single value, such as that stored in a field of a larger **record** in a database. It is the smallest useful unit of named information in the system.
derive, derives	To retrieve from a specified source.
descriptive data structure	A machine-readable data structure (e.g., text file, template, **secure container**, etc.) **containing** or **addressing** descriptive information (e.g., **metadata**, shorthand abstract representation, integrity constraints, **rules**, instructions, etc.) about (1) the layout, generic format (e.g., location of a particular type of information), attributes, or hierarchical structure (e.g., file hierarchy) of the contents section of one or a family of other data structure(s) (e.g., **secure container**, other rights management related structure, etc.), (2) the operations or processes used to create or **use** such other data structure(s) (e.g., **rules** for handling the data structure), and/or (3) the consequences of such operations (e.g., billing the user a certain fee for printing). The descriptive data structure is capable of being used to create or handle (e.g., read, locate information within, request information from, and/or manipulate) the other data structure(s). The descriptive data structure is not **associated with** the other data structure(s) and does not **contain** or specify its particular contents (e.g., "Yankees Win the Pennant!").
designating	[This shall be construed in connection with a disputed claim phrase.]
device class	The generic name for a group of device types. For example, all display stations belong to the same device class. A device class is different from a device type. A device type is composed of all devices that share a common model number or family (e.g. IBM 4331 printers).
digital file	A static unit of storage allocated by a "**file system**" and **containing** digital information. A digital file enables any application using the "**file system**" to randomly access its contents and to distinguish it by name from every other such unit. A **copy** of a digital file is a separate digital file. (A "file system" is the portion of the operating system that translates requests made by application programs for operations on "files" into low-level tasks that can control storage devices such as disk drives.)
digital signature, digitally signing	digital signature: An unforgable string of characters (e.g., bits) generated by a cryptographic transformation to a block of data using some secret, which string can be generated only by an agent that knows the secret, and hence provides evidence that the agent must have generated it. digitally signing: Creating a digital signature using a secret **key**. (In symmetric key cryptography, a "secret key" is a **key** that is known only to the sender and recipient. In asymmertric key cryptography, a "secret key" is the private key of a public/private key pair, in which the two keys are related uniquely by a predetermined mathematical relationship such that it is computationally infeasible to determine one from the other.)
entity, entity's control	entity: Any person or organization.

Claim Term	Preliminary Construction
	entity's control: **Control** created, modified, or selected by any person or organization to **control** a particular **use** of or **access** to particular protected information by a particular user(s).
environment	[This will be construed in connection with other disputed claim terms.]
executable programming, executable (2)	executable: A cohesive series of machine code instructions in a format that can be loaded into memory and run (executed) by a connected processor. executable programming: A cohesive series of machine code instructions, comprising a **computer program**, in a format that can be loaded into memory and run (executed) by a connected processor. (A "computer program" is a complete series of definitions and instructions that when executed on a computer will perform a required or requested task.)
execution space, execution space identifier	execution space: A processor-addressable physical **memory** into which data and executable code can be loaded, which is assigned to a single executing process while that process is actively executing. **Memory** holding "swapped out" processes or **executables** is not part of an "execution space." execution space identifier: A value that uniquely **identifies** a particular execution space.
generating	[This shall be construed in connection with a disputed claim phrase.]
govern, governed, governed item, governing	govern, governing, governed: See **control** (v.) governed item: Information, of arbitrarily fine granularity, whose access and use by any user, process, or device which is **controlled**.
halting	Stopping execution of a running (executing) process unconditionally (i.e., without providing any specific condition for resumption). For example, executing an instruction known as a "breakpoint halt instruction."
host processing environment	A **processing environment** within a **VDE** node which is not a **Secure Processing Environment**. A "host processing environment" may either be "secure" or "not secure." A "secure" host processing environment is a self-contained **protected processing environment**, formed by loaded, **executable programming** executing on a general purpose CPU (not a **Secure Processing Unit**) running in protected (privileged) mode. A "non-secure" host processing environment is formed by loaded, **executable programming** executing on a general purpose CPU (not a **Secure Processing Unit**) running in user mode.
identifier, identify, identifying	identifier: Any text string used as a label naming an individual instance of what it identifies. identify: To establish as being a particular instance of a person or thing.
including	(With respect to a **digital file**, **control**, **authorization information**, **Secure Processing Environment**, **descriptive data structure**, element, **load module**, header, or **secure container**): Physically storing within, as opposed to **addressing**.
information previously stored	Information that once was stored but is no longer stored.

Claim Term	Preliminary Construction
integrity programming	**Executable programming** that when executed checks and reports on the **integrity** of a device or process. "Integrity" means the property that information has not been altered either intentionally or accidentally.
key	A bit sequence used and needed by a cryptographic algorithm to encrypt a block of plain text or to decrypt a block of cipher text. A key is different from a key seed or other information from which the actual encryption and/or decryption key is constructed, **derived**, or otherwise identified. In symmetric key cryptography, the same key is used for both encryption and decryption. In asymmetric or "public key" cryptography, two related keys are used; a block of text encrypted by one of the two keys (e.g., the "public key") can be decrypted only by the corresponding key (e.g., the "private key").
load module (2)	An executable, modular unit of machine code suitable for loading into **memory** for execution by a processor. A load module is encrypted (when not within a **Secure Processing Unit**) and has an **identifier** that a calling process must provide to be able to use the load module. A load module is combinable with other load modules, and associated data, to form executable **component assemblies**. A load module can execute only in a **VDE protected processing environment**.
machine check programming	**Executable programming** that when executed generates a unique "machine signature" which distinguishes the physical machine from all other machines. This machine check programming code sometimes is invoked by **integrity programming**.
metadata information (2 (metadata))	Data that describes other data managed within an application or environment, such as its meaning, representation in storage, what it is used for and by whom, context, quality and condition, or other characteristics. Metadata may describe data elements or attributes (name, size, data type, etc) and data about **records** or data structures (length, fields, columns, etc) and data about data (where it is located, how it is associated, ownership, etc.).
opening secure containers	Establishing the requisites needed to attempt to **access** the contents of a **secure container**. Opening is a necessary but insufficient step before the contents of a **secure container** may be copied, decrypted, read, manipulated, or otherwise **used**, or **accessed**. No process, user, or device may **access** or **use** the contents of a **secure container** without first opening that **secure container**. A **secure container** may be opened only through execution of the assigned **VDE control**(s) within a **VDE Secure Processing Environment** and satisfaction of all requirements imposed by such execution.
operating environment	See **processing environment**.
organization, organization information, organize	organization, organization information: The manner in which data is represented and laid out in physical storage. For example, for data organized as records: the field hierarchy, order, type and size. organize: Representing and laying out data in a particular manner in physical storage.
portion	[This shall be construed in connection with a disputed claim phrase.]
prevents	Imposes an active restraint on an action such that it absolutely cannot occur by any means or under any circumstances.
processing environment (2 – 912:35, 900:155, 721:34)	A standardized, well-defined, self-contained, computing base, formed by hardware and executing code, that provides an "interface" and set of resources which can support different applications, on different types of hardware platforms. In the context of claim 35 of the '912 patent: a **Secure Processing**

Claim Term	Preliminary Construction
	Environment.
protected processing environment (2 – 721:34)	1. A uniquely identifiable, self-contained computing base trusted by all VDE nodes to protect the **availability, secrecy, integrity** and **authenticity** of all information identified in the patent application as being protected, and to guarantee that such information will be **accessed** and **used** only as expressly **authorized** by **VDE controls**. At most VDE nodes, the protected processing environment is a **Secure Processing Environment** which is formed by, and requires, a hardware **tamper-resistant barrier encapsulating** a special-purpose **Secure Processing Unit** having a processor and internal **secure memory**. ("Encapsulated" means hidden within an object so that it is not directly accessible but rather is accessible only through the object's restrictive interface.) The **barrier prevents** all **unauthorized** (intentional or accidental) interference, removal, observation, and **use** of the information and processes within it, by all parties (including all users of the device in which the Protected Processing Environment resides), except as expressly **authorized** by **VDE controls**. A Protected Processing Environment is under **control** of **controls** and **control information** provided by one or more parties, rather than being under **control** of the appliance's users or programs. Where a VDE node is an established financial clearinghouse, or other such facility employing physical facility and user-identity authentication security procedures trusted by all VDE nodes, and the VDE node does not **access** or **use** VDE-protected information, or assign VDE **control information**, then the Protected Processing Environment at that VDE node may instead be formed by a general-purpose CPU that executes all VDE "security" processes in protected (privileged) mode. 2. A Protected Processing Environment requires more than just verifying the **integrity** of digitally signed **executable programming** prior to execution of the programming; or concealment of the program, associated data, and execution of the program code; or use of a password as its protection mechanism.
protecting	Maintaining the **security** of.
record (n.)(2)	A data structure that is a collection of fields (elements), each with its own name and type. Unlike an array, whose elements are accessed using an index, the elements of a record are accessed by name. A record can be accessed as a collective unit of elements, or the elements can be accessed individually.
required	A condition without which an action cannot occur. A required condition acts prospectively – it does not apply to a description created at or after the creation of the object to which it applies.
resource processed	A **record** containing **control information**, which **record** is stored and acted upon within a **processing environment.**
rule (2)	A lexical statement that states a condition under which **access** to or **use** of VDE-protected data will be **allowed** by a **VDE control**. A rule may specify how, when, where, and by whom a particular activity on particular information is to be **allowed**.
secure (2)	A state in which all users of a system are guaranteed that all information, processes, and devices within the system, shall have their **availability, secrecy, integrity** and **authenticity** maintained against all of the identified threats thereto. "Availability" means the property that information is accessible and usable upon demand by authorized persons, at least to the extent that no user may delete the information without **authorization**. "Secrecy," also referred to as confidentiality, means the property that information (including computer processes) is not made available or disclosed to unauthorized persons or processes. "Integrity" means the property that information has not been altered either intentionally or accidentally. "Authenticity" means the

Claim Term	Preliminary Construction
	property that the characteristics asserted about a person, device, program, information, or process are genuine and timely, particularly as to identity, data integrity, and origin integrity.
secure container	A VDE secure container is a self-contained, self-protecting data structure which (a) **encapsulates** information of arbitrary size, type, format, and **organization**, including other, nested, containers, (b) cryptographically protects that information from all **unauthorized access** and **use**, (c) provides encrypted storage management functions for that information, such as hiding the physical storage location(s) of its protected contents, (d) permits the **association** of itself or its contents with **controls** and **control information governing access to** and **use** thereof, and (e) **prevents** such **use** or **access** (as opposed to merely **preventing** decryption) until it is "opened." A secure container can be opened only as expressly **allowed** by the **associated VDE control(s)**, only within a **Secure Processing Environment**, and only through decryption of its encrypted header. A secure container is not directly accessible to any non-VDE or user calling process. All such calls are intercepted by VDE. The creator of a secure container can assign (or allow others to assign) **control information** to any arbitrary portion of a secure container's contents, or to an empty secure container (to **govern** the later addition of contents to the container, and **access** to or **use** of those contents). A container is not a secure container merely because its contents are encrypted and signed. A secure container is itself **secure**. All VDE-protected information (including protected content, information about content usage, and content-**control information**, **controls**, and **load module**) is **encapsulated** within a secure container whenever stored outside a **Secure Processing Environment** or **secure database**.
secure container governed item	A **governed item protected** by a **secure container**. A secure container governed item may not be **accessed** or **used** in any way, by any user, process, or device, except as **allowed** by its **associated VDE control(s)** executing in a **VDE Secure Processing Environment** and satisfaction of all requirements imposed by such execution.
secure container rule	A **rule protected** by a **secure container**. A secure container rule may not be **accessed** or **used** in any way, by any user, process, or device, except as **allowed** by its **associated VDE control(s)** executing in a **VDE Secure Processing Environment** and satisfaction of all requirements imposed by such execution.
secure database	A data store isolated from all users such that it is **protected** from external observation; and accidental or intentional alteration or destruction. In VDE, a secure database stores tracking, billing, payment, and **auditing** data until the data is delivered **securely** to an authorized **clearinghouse**.
secure execution space	An allocated portion of the **secure memory** within a special-purpose **Secure Processing Unit** which is isolated from the rest of the world, and **protected** from observation by (and **encapsulated** within) a **tamper resistant barrier** and **protected** from alteration by the processor. The processor cryptographically verifies the **integrity** of all code loaded from secure memory prior to execution, executes only the code that the processor has **authenticated** for its use, and is otherwise **secure**.
secure memory, memory	memory: A medium in which data (including executable instructions) may be stored and from which it may be retrieved. "Memory" does not include a "virtual memory." secure memory: A processor-addressable **memory** within a special-purpose **Secure Processing Unit** which is isolated from the rest of the world by (and **encapsulated** within) a **tamper resistant barrier**. "Processor-addressable" means that a connected processor can use the secure memory's

Claim Term	Preliminary Construction
	physical addresses as the operand in a processor instruction such as LOAD or STORE or equivalent instruction. A "memory" is not a "secure memory" merely because it stores encrypted, signed, and/or sealed data; is accessible from a **Protected Processing Environment**; or is within an appliance that is located at a trusted facility with non-VDE physical security and user-identity authentication procedures.
secure operating environment, said operating environment	Same as **Secure Processing Environment**.
securely applying (2 – securely)	securely: Performed in a **Secure Processing Environment** in a manner that guarantees that each affected information or process remains **secure**. securely applying: **securely** (1) executing the applied **executables** (e.g., **controls**) within a **VDE secure execution space**, (2) validating and verifying the **authenticity** and **integrity** of each **executable**, and (3) ensuring that the **executables** are applied only in ways that are intended by the **VDE** participants who created the **executables**.
securely assembling	**securely** (1) linking or binding plural distinct elements together in a particular manner (specified by **authenticated** assembly instructions) into a single cohesive executable unit so the elements can directly reference each other element within the resulting assembly, within a **VDE Secure Processing Environment**, (2) validating and verifying the **authenticity** and **integrity** of each element (e.g., that it has not been modified from or substituted for the correct element) immediately prior to binding it into the assembly, and (3) ensuring that the elements are linked together only in ways that are intended by the **VDE** participants who created the elements and/or specified the assembly thereof.
securely processing	Executing code in a **secure execution space** to act upon some information, in a manner that ensures that the information and the processing remain **secure**.
securely receiving	Receiving digital information in a **secure container**, as part of a communication encrypted on the communications level, at a **Secure Processing Environment authenticated** in accordance with **VDE controls associated with the secure container**.
security (2)	See **secure**.
security level, level of security	An ordered measure of the degree of **security**. The "security level" is persistent unless expressly noted to exist only some of the time. Also, the combination of a hierarchical classification and a set of nonhierarchical categories that represents the sensitivity of an object or the clearance of a subject. For example, Unclassified, Confidential, Secret, and Top Secret are hierarchical classifications, whereas NATO and NOFORN are non-hierarchical categories defined by the DoD Trusted Computing guidelines.
specific information, specified information	[This will be construed in connection with disputed claim phrases]
tamper resistance (2 – tamper)	tamper resistance: The ability of a **tamper resistant barrier** to **prevent access**, observation, and interference with information or processing **encapsulated** by the barrier.

Claim Term	Preliminary Construction
	tamper: See **tampering.**
tamper resistant barrier	An active device that **encapsulates** and separates a **Protected Processing Environment** from the rest of the world. It prevents information and processes within the **Protected Processing Environment** from being observed, interfered with, and leaving except under appropriate conditions ensuring **security**. It also **controls** external **access** to the **encapsulated secure** resources, processes and information. A **tamper resistant barrier** is capable of destroying **protected** information in response to **tampering** attempts.
tamper resistant software	Software that is **encapsulated** and executed wholly within a **tamper resistant barrier.**
tampering (2)	Attempting to circumvent a **tamper resistant barrier** or other mechanism designed to protect against the observation, **access**, or alteration of data, code, or process execution, or making any unauthorized access, observation, or interference.
use (n.)	Any action with respect to information (e.g., copying, printing, decrypting, executing) other than **access**. In VDE, information use is **allowed** only through execution of the applicable **VDE control**(s) and satisfaction of all requirements imposed by such execution.
user controls (1)	**Controls** created, modified, or selected by a user to control a particular **use** or **access** by the user to particular protected information.
validity	The state in which **authenticated** data conforms to predetermined completeness and consistency parameters.
virtual distribution environment	See Global Construction of **VDE.**

Microsoft PLR 4-2, Exhibit C: Mini Markman Preliminary Claim Construction -- Claim Phrases[1]

Claim Phrase	Microsoft's Preliminary Construction
'193:1	Claim as a Whole: The recited method is performed within a **VDE**.
receiving a digital file including music	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The qualifier "including music" recites a non-functional descriptive material and is not a patentable limitation. 3. The recited function requires: obtaining a **VDE secure container encapsulating a digital file, authenticating** the intended recipient in accordance with **VDE controls associated with** the **secure container**, and accepting the **secure container**.
a budget specifying the number of copies which can be made of said digital file	1. A **budget** identifying the total number of **copies** (whether or not decrypted, long-lived, or accessible) that (since creation of the budget) can be made of the **digital file** by any and all users, devices, and processes. No process, user, or device is able to make another **copy** of the **digital file** once this number of **copies** has been made.
controlling the copies made of said digital file	1. **Controlling uses** of and **accesses** to all **copies** of the **digital file**, by all users, processes, and devices, by executing each of the recited "at least one" **copy control**(s) within **VDE Secure Processing Environment**(s). Each **control governs (controls)** only one action, which action may or may not differ among the different "at least one" **controls**. All **uses** and **accesses** are prohibited and incapable of occurring except to the extent **allowed** by the "at least one" **copy control**(s).
determining whether said digital file may be copied and stored on a second device based on at least said copy control	1. Determining whether this particular first device is **allowed** to perform both of the following actions on this particular **digital file**: (1) **copy** it and (2) store it (as opposed to a **copy** of it) on a second device, by executing one or more **VDE control**(s) (including "said" **copy control associated with** this **digital file**) within **VDE Secure Processing Environment**(s). To the extent that either of these two actions is not determined by this step to be permissible, that action is absolutely prohibited and incapable of occurring, and no user, process or device can perform it on this **digital file**.

[1] The word "invention" is used not to suggest that anything described in InterTrust's patents in fact was novel or non-obvious or inventive, but rather to identify what was described as the **alleged** invention. Also, features and capabilities are described as they are described in the InterTrust patent application, even though the patent application did not describe an actual working system having any of these capabilities. Also, Microsoft's proposed constructions use many terms from the InterTrust patents that are used inconsistently or otherwise indefinitely in the patents. Those terms are used by Microsoft in their narrowest applicable sense, and without waiving the right to assert the indefiniteness of this claim language. Also, the preliminary constructions assume (without conceding) that the February, 1995, InterTrust patent application was incorporated by reference into the '721, '861, and '683 patents, effectively for claim construction purposes. If the Court concludes otherwise, then the proper constructions will be different in some cases. Bolded terms are preliminarily defined in Exhibits A-C of Microsoft's PLR 4-2 papers.

Claim Phrase	Microsoft's Preliminary Construction
	2. This claim limitation's recitation of "said" **copy control** is inconsistent with the claim limitation "at least one" **copy control**.
if said copy control allows at least a portion of said digital file to be copied and stored on a second device	1. This "if" condition creates two branches for the recited process, each of which must be performed. Each time the "if" condition is met, all four of the later-recited actions (copying, transferring, storing, playing) must occur. Each time it is not met, each of these four actions must be prohibited and incapable of occurring. 2. This "if" condition is met if and only if "said" **copy control allows** any portion of the **digital file** to be **copied** and also **allows** that same **portion** of the **file** (as opposed to the **copy**) to be stored on any second device. This "if" condition is based entirely on "said copy control" and thus is met, as above, even if other **VDE control(s)** prohibit those actions. 3. This claim limitation's recitation of "copy control allows at least a portion" is inconsistent with the claim limitation "whether said digital file may be copied … based on at least said copy control."
copying at least a portion of said digital file	1. **Copying** at least some portion of the **digital file** (as opposed to a **copy** thereof), by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**. This copied "portion" may or may not be (or even include) the portion referred to in the claim limitation "if said copy control allows at least a portion."
transferring at least a portion of said digital file to a second device	1. Transferring to some second device (which may or may not be the "second device" referred to in the claim limitation "if said copy control allows at least a portion of said digital file to be copied and stored on a second device") at least some portion of the **digital file** (as opposed to a **copy** thereof), by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**. This transferred portion may or may not be (or even include) the portion referred to in the claim limitation "if said copy control allows at least a portion," or the portion referred to in the claim limitation "copying at least a portion."
storing said digital file	1. Storing the entire **digital file** received in the "receiving" step (as opposed to a **copy** of the **file** or an incomplete portion of the **file**). 2. This claim limitation's recitation of "storing said digital file" is inconsistent with the claim limitation "transferring at least a portion of said digital file."
'193:11	Claim as a Whole: The recited method is performed within a **VDE**.
receiving a digital file	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: obtaining a **VDE secure container encapsulating a digital file, authenticating** the intended recipient in accordance with **VDE controls associated with** the **secure container**, and accepting the **secure container**.
determining whether said digital file may be copied	1. Determining whether said first **control**, by itself, **allows** this particular first device to perform both of the following actions on this particular **digital file**: (1) **copy** it and (2) store it (as opposed to **a copy** of it) on a second device, by executing the first **VDE control** within **VDE Secure Processing**

Claim Phrase	Microsoft's Preliminary Construction
and stored on a second device based on said first control	**Environment(s)**. To the extent that either the **copy** or store action is not determined by this step to be permissible, that action is absolutely prohibited and incapable of occurring, and no user, process or device can perform it on this **digital file**.
identifying said second device	1. Identifying a second device sufficiently to distinguish it from all other devices, by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**.
whether said first control allows transfer of said copied file to said second device	1. Whether the first **control**, by itself, **allows** the entire **digital file** (which has been copied at least once) (as opposed to the copy) to be moved to the identified second device. If not, that transfer is absolutely prohibited and incapable of occurring and no user, process or device can perform that action on this **file**.
said determination based at least in part on the features present at the device	1. Basing the determination at least in part upon all actual, current features of the device (as opposed to previously determined, reported, or measured features) which might affect the device's ability to prevent unauthorized **access** to and/or **use** of the **digital file**. This determination is done without trusting either the device or any user of the device. A device identifier such as a serial number is not a "feature present at the device."
if said first control allows at least a portion of said digital file to be copied and stored on a second device	1. This "if" condition creates two branches for the recited process, each of which must be performed. Each time the "if" condition is met, all four of the later-recited actions (copying, transferring, storing, rendering) must occur. Each time it is not met, each of these four actions must be disabled and prohibited and incapable of occurring. 2. This "if" condition is met if and only if the first **control** allows any portion of the **digital file** to be **copied** and also allows that same **portion** of the **file** (as opposed to the copy) to be on any second device. This "if" condition is based entirely on the first **control** and thus is met, as above, even if other **VDE controls** prohibit those actions. 3. This claim limitation's recitation of "said first control allows at least a portion" is inconsistent with the claim limitation "whether said digital file may be copied ... based on said first control."
copying at least a portion of said digital file	1. **Copying** at least some portion of the **digital file** (as opposed to a **copy** thereof), by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**. The copied portion may or may not be (or even include) the portion referred to in the claim limitation "if said first control allows at least a portion."
transferring at least a portion of said digital file to a second device	1. Transferring to some second device (which may or may not be the "second device" referred to in the claim limitation "if said first control allows at least a portion of said digital file to be copied and stored on a second device") at least some portion of the **digital file** (as opposed to a **copy** thereof), by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**. The transferred portion may or may not be (or even include) the portion referred to in the claim limitation "if said first control allows at least a portion," or the portion referred to in the claim limitation "copying at least a portion."
storing said digital file	1. Storing the entire **digital file** received in the "receiving" step (as opposed to a **copy** of the **file** or an incomplete portion of the **file**).

Claim Phrase	Microsoft's Preliminary Construction
	2. This claim limitation's recitation of "storing said digital file" is inconsistent with the claim limitation "transferring at least a portion of said digital file."
'193:15	Claim as a Whole: The recited method is performed within a **VDE**.
receiving a digital file	1. See 193:11. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step).
an authentication step comprising:	1. **Authenticating** the first device and/or user of the first device without relying on trusting either, by executing **VDE control**(s) within **VDE Secure Processing Environment**(s).
accessing at least one identifier associated with a first device or with a user of said first device	1. **Securely accessing** at least one **identifier associated with** a single ("first") device or (as opposed to "and") with a single, current user of that device, by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). One of the "at least one identifier" may be **associated with** a first device while another of the "at least one identifier" may be **associated with** a user of said first device.
determining whether said identifier is associated with a device and/or user authorized to store said digital file	1. For each accessed "at least one **identifier**," determining whether the device with which it is **associated** is one on which the **file** may be stored (by any user) and/or whether the user with which it is **associated** is one who may store the **file** (on any device), by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). Each **identifier** may be **associated with** a device "and" a user, or with a device only, or with a user only. 2. This claim limitation's recitation of "said identifier" is inconsistent with the claim limitation "at least one identifier."
storing said digital file in a first secure memory of said first device, but only if said device and/or user is so authorized, but not proceeding with said storing if said device and/or user is not authorized	1. This conditional step creates at least two "**authentication**" branches for the recited process, each of which must be performed. Each time the condition is met, the recited "storing" must occur. Each time it is not met, the recited "storing" must not occur. 2. If "storing" proceeds, then: storing in a **secure memory** of the first device, the entire **file** received in the "receiving" step, as opposed to a **copy** of the **file** or an incomplete portion of the **file**, by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). If "storing" does not proceed: then the **file** is not stored in the **secure memory** of the first device, and is absolutely prevented from being stored anywhere on the first device. 3. This limitation is internally inconsistent on the circumstances under which the storing proceeds or does not proceed. For example, the first ("only if") phrase requires that the storing step proceeds if the device is authorized (and the user is not) while the second ("but not") phrase requires that the storing step not proceed if the device is authorized (and the user is not).
storing information associated with said digital file in a secure database stored on said first device, said information including	1. Storing information in a **secure database**, the entirety of information (including the "at least one **control**") being **associated with** the **digital file** (as opposed to the **file's** contents independent of the **file**), by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). 2. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step).

Claim Phrase	Microsoft's Preliminary Construction
at least one control	
determining whether said digital file may be copied and stored on a second device based on said at least one control	1. Determining whether the "at least one control," by itself or themselves, **allow**(s) this particular first device to perform both of the following actions on this particular **digital file**: (1) **copy** it and (2) store it (as opposed to **a copy** of it) on a second device, by executing "said at least one control," by executing the "at least one" **VDE control** within **VDE Secure Processing Environment**(s). To the extent that either the **copy** or store action is not determined by this step to be permissible, that action is absolutely prohibited and incapable of occurring, and no user, process or device can perform it on this **digital file**. 2. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step).
if said at least one control allows at least a portion of said digital file to be copied and stored on a second device,	1. This "if" condition creates two branches for each of the two "authentication branches" of the recited process (and thus four branches in all), each of which must be performed. Each time it is met, all four of the later-recited actions (copying, transferring, storing, rendering) must occur. Each time it is not met, each of these four actions must be prohibited and incapable of occurring. 2. This "if" condition is met if and only if the at least one **control allows** any portion of the **digital file** to be **copied** and also allows that same portion of the **file** (as opposed to the **copy**) to be stored on any second device. This "if" condition is based entirely on the at least one **control** and thus is met, as above, even if other VDE **controls** prohibit those actions. 3. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step). 4. This claim limitation's recitation of "at least one control allows at least a portion of said digital file" is inconsistent with the claim limitation "whether said digital file may be copied ... based on said at least one control."
copying at least a portion of said digital file	1. Copying at least some portion of the **digital file** (as opposed to a **copy** thereof), which portion may or may not be (or even include) the portion referred to in the claim limitation "if said at least one control allows at least a portion," by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). 2. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step).
transferring at least a portion of said digital file to a second device	1. Transferring to some second device (which may or may not be the "second device" referred to in the claim limitation "if said at least one control allows at least a portion of said digital file to be copied and stored on a second device") at least some portion of the **digital file** (not a copy thereof), by executing **VDE control**(s) within **VDE Secure Processing Environment**(s). The transferred portion may or may not be (or even include) the portion referred to in the claim limitation "if said at least one control allows at least a portion," or the portion referred to the claim limitation "copying at least a portion." 2. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step).
storing said digital file	1. Storing the entire **digital file** received in the "receiving" step (as opposed to a **copy** of the **file** or an incomplete portion of the **file**).

Claim Phrase	Microsoft's Preliminary Construction
	2. This step must proceed in both "**authentication** branches" of the process (i.e., regardless of the outcome of the "**authentication**" step). 3. This claim limitation's recitation of "storing said digital file" is inconsistent with the claim limitation "transferring at least a portion of said digital file."
'193:19	Claim as a Whole: The recited method is performed within a **VDE**.
receiving a digital file at a first device	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: obtaining a **VDE secure container encapsulating a digital file, authenticating** the first device in accordance with **VDE controls associated with** the **secure container**, and accepting the **secure container**.
establishing communication between said first device and a clearinghouse located at a location remote from said first device	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("establishing communication") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step Part of the recited function is performed by the Remote Procedure Call Manager 732 software of Rights Operating System 602 that controls I/O controller 660 and Communications Controller 666. 2. The recited function is: creating and using a previously non-existent communications channel which is necessary and sufficient for exchanging information between the first device and a **clearinghouse**.
using said authorization information to gain access to or make at least one use of said first digital file	1. A user, process or device uses all of said **authorization information** in connection with executing **VDE control**(s) within **VDE Secure Processing Environment**(s) to gain **access** to or (as opposed to "and") make at least one **use** of the **file** received in the "receiving" step. Without using such authorization information, no **access** to or **use** of the file is **allowed**.
including using said key to decrypt at least a portion of said first digital file	1. The "at least one use of said digital file" must encompass decrypting at least a portion of the **digital file** using the **key**.
receiving a first control from said clearinghouse at said first device	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: obtaining a **VDE secure container encapsulating a first control, authenticating** the first device in accordance with **VDE controls associated with the secure container**, and accepting the **secure container**.
storing said first digital file	1. Storing in a **memory** of the first device, the entire **digital file** (as opposed to any incomplete portion thereof) received in the "receiving" step, by

Claim Phrase	Microsoft's Preliminary Construction
in a memory of said first device	executing **VDE control(s)** within **VDE Secure Processing Environment(s)**.
using said first control to determine whether said first digital file may be copied and stored on a second device	1. Determining whether the first **control**, by itself, allows this particular first device to perform both of the following actions on this particular **digital file**: (1) **copy** it and (2) store it (as opposed to **a copy** of it) on a second device, by executing the first **VDE control** within **VDE Secure Processing Environment(s)**. To the extent that either the **copy** or store action is not determined by this step to be permissible, that action is absolutely prohibited and incapable of occurring, and no user, process or device can perform it on this **digital file**.
if said first control allows at least a portion of said first digital file to be copied and stored on a second device	1. This "if" condition creates two branches for the recited process, each of which must be performed. Each time the "if" condition is met, all four of the later-recited actions (copying, transferring, storing, rendering) must occur. Each time it is not met, each of these four actions must be prohibited and incapable of occurring. 2. This "if" condition is met if and only if the first **control** allows any portion of the first **digital file** to be **copied** and also allows that same **portion** of the **file** (as opposed to the copy) to be stored on any second device. This "if" condition is based entirely on the first control and thus is met, as above, even if other VDE **controls** prohibit those actions. 3. This claim limitation's recitation of "first control allows at least a portion of said first digital file" is inconsistent with the claim limitation "whether said first digital file may be copied … on a second device."
copying at least a portion of said first digital file	1. **Copying** at least some portion of the **digital file** (as opposed to a **copy** thereof), which portion may or may not be (or even include) the portion referred to in the claim limitation "if said first control allows at least a portion," by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**.
transferring at least a portion of said first digital file to a second device including a memory and an audio and/or video output	1. Transferring to some second device (which may or may not be the "second device" referred to in the claim limitation "if said first control allows at least a portion of said digital file to be copied and stored on a second device") at least some portion of the **digital file** (not a copy thereof), by executing **VDE control(s)** within **VDE Secure Processing Environment(s)**. The transferred portion may or may not be (or even include) the portion referred to in the claim limitation "if said first control allows at least a portion," or the portion referred to the above limitation "copying at least a portion."
storing said first digital file portion	1. Storing the "at least a portion" which was transferred to the second device, of the **digital file** received in the "receiving" step (as opposed to a **copy** of the **file**).
'683:2	Claim as a Whole: The "system" is a **VDE**.
user controls	1. [This shall be construed as a disputed claim term.]
the first secure container having been received from a	1. The "first secure container" must **identify** the single apparatus from which it was received, and that apparatus must be different from the first apparatus. Alternatively, if the Court does not construe this claim language as requiring the "first secure container" to identify the single apparatus

Claim Phrase	Microsoft's Preliminary Construction
second apparatus	from which it was received: This claim language has no patentable weight. It recites a step taken in the creation of the recited system, not a structural or functional characteristic of the system. One studying a particular system (as opposed to the process by which it was created) to compare it to the claimed system, could not distinguish a **secure container** received from another apparatus from, e.g., a **secure container** created on the first apparatus, and thus could not determine whether this step was satisfied. 2. Receiving the **secure container** includes **authenticating** the intended recipient in accordance with **VDE controls associated with** the **secure container**. The first **secure container** may be received as bar codes in a fax transmission, or filled ovals on a form delivered through physical mail.
an aspect of access to or use of	1. Any one (as opposed to more than one) aspect of any **access** to or (as opposed to "and") **use** by any and all processes, users, and devices.
the first secure container rule having been received from a third apparatus different from said second apparatus	1. The "first secure container rule" must have been received **encapsulated** within a **VDE secure container**, and the intended recipient must have been **authenticated** in accordance with **VDE controls associated with** the **secure container**, and the "first secure container rule" must have been accepted by the first apparatus. The "first secure container rule" must identify the single apparatus from which it was received, and that apparatus must be different from the first apparatus. 2. Alternatively, if the Court does not construe this claim language as requiring the "first secure container" to identify the single apparatus from which it was received: This claim language has no patentable weight. It recites a step taken in the creation of the recited system, not a structural or functional characteristic of the system. One studying a particular system (as opposed to the process by which it was created) to compare it to the claimed system, could not distinguish a **secure container rule** received from another apparatus from, e.g., a **secure container rule** created on the first apparatus, and thus could not determine whether this step was satisfied.
hardware or software used for receiving and opening secure containers	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites an undefined mechanism ("hardware or software") for performing a function (e.g., "**opening**") without reciting particular structure that performs that function. The specification does not clearly link any particular structure to this recited function. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: the same single logical piece of either hardware or software (as opposed to both) must be capable of both receiving and **opening secure containers**, this "receiving" including **authenticating** the intended recipient in accordance with **VDE controls associated with** the **secure container**, and this "opening" performed by executing **VDE control**(s) within **VDE Secure Processing Environment**(s).
said secure containers each including the capacity to contain a governed item, a secure container rule being associated with each of said secure containers	1. Each **secure container** which the "hardware or software used for receiving and opening secure containers" is capable of receiving and opening must have the capacity to **contain a governed item**, and must have **associated with** it (as opposed to any particular **governed item**) a **secure container rule**.

Claim Phrase	Microsoft's Preliminary Construction
protected processing environment at least in part protecting information contained in said protected processing environment from tampering by a user of said first apparatus	1. A single **VDE Secure Processing Environment**, in addition to and not within the first apparatus, actively **preventing** (not merely being capable of **preventing**, and not merely resisting) any "user" of the first apparatus from **tampering** with any and all information **encapsulated** by the **Secure Processing Environment** (as opposed to **tampering** with the **Secure Processing Environment** itself). Other components may or may not provide part of this **protecting** function. 2. The **protecting** function is provided by use of the disclosed "**component assembly**" (VDE **controls**), "**secure container**," "**protected processing environment**," "object registration," and other mechanisms of the purported "**VDE**" "invention" for allegedly individually ensuring the "access control" "handcuffs" between specific "**controls**," specific "objects" (and their content at an arbitrary granular level), and specific "users."
hardware or software used for applying said first secure container rule and a second secure container rule in combination to at least in part govern at least one aspect of access to or use of a governed item contained in a secure container	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites an undefined mechanism ("hardware or software") for performing a function ("applying ... in combination") without reciting particular structure that performs that function. The specification does not clearly link any particular structure to this recited function. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: a single logical piece of either hardware or software (as opposed to both) to apply the two separate **rules** in combination by assembling and executing a single **control**, and to **govern** any one or more aspects of any **access** or **use** by any process or user or device, of a **governed item contained** in a **secure container** (which may or may not be any "secure container" recited earlier). Other components may or may not provide part of the **governing** function. This "hardware or software" performs its functions by executing **VDE control**(s) within **VDE Secure Processing Environment**(s).
hardware or software used for transmission of secure containers to other apparatuses or for the receipt of secure containers from other apparatuses.	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites an undefined mechanism ("hardware or software") for performing a function (e.g., "transmission") without reciting particular structure that performs that function. The specification does not clearly link any particular structure to this recited function. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: a single logical piece of either hardware or software (as opposed to both) is capable of both transmission and receipt of **secure containers**, this receipt including **authenticating** the intended recipient in accordance with **VDE controls associated with** the **secure container**. This "hardware or software" is separate from and in addition to the first apparatus, the recited **protected processing environment**, and the recited "hardware or software used for receiving and opening secure containers." The transmission and receipt of the **secure containers** may be via bar codes in a fax transmission, or filled ovals on a form delivered through physical mail. This "hardware or software" performs its functions by executing **VDE control**(s) within **VDE Secure Processing Environment**(s).
'721:1	Claim as a Whole: The recited method is performed within a **VDE**.
digitally signing a first load module with a first digital signature designating the	1. **Digitally signing** a particular ("first") **load module** by using a first **digital signature** as the signature **key**; which signing indicates to any and all devices in the first device class that the signor authorized this **load module** for use by that device. No **VDE** device can perform any execution of any **load module** without such authorization. The method ensures that the **load module** cannot execute in a particular device class and ensures that no

Claim Phrase	Microsoft's Preliminary Construction
first load module for use by a first device class	device in that device class has the key(s) necessary to verify the **digital signature.**
digitally signing a second load module with a second digital signature different from the first digital signature, the second digital signature designating the second load module for use by a second device class having at least one of tamper resistance and security level different from the at least one of tamper resistance and security level of the first device class	1. **Digitally signing** a different ("second") **load module** by using a different ("second") **digital signature** as the signature key, which signing indicates to any and all devices in the second device class that the signor authorized this **load module** for use by that device. No **VDE** device can perform any execution of any **load module** without such authorization. The method ensures that the **load module** cannot execute in a particular device class and ensures that no device in that device class has the key(s) necessary to verify the digital signature. 2. All devices in the first device class have the same persistent (not just occasional) and identified level of **tamper resistance** and/or same persistent and identified **level of security**. All devices in the second device class have the same persistent and identified level of **tamper resistance** and/or same persistent and identified **level of security**. The identified level of **tamper resistance** and/or identified **level of security** for the first device class, is greater or less than the identified **level of tamper resistance** and/or identified **level of security** for the second device class.
distributing the first load module for use by at least one device in the first device class	1. The first **load module, digitally signed** as indicated above, is transmitted to at least one device in the first device class.
distributing the second load module for use by at least one device in the second device class	1. The second **load module, digitally signed** as indicated above, is transmitted to at least one deviee in the second device class.
'721:34	Claim as a Whole: The "protected processing environment" is part of and within **VDE.**
arrangement within the first tamper resistant barrier	1. The **arrangement** is located and executed wholly within the first **tamper resistant barrier.**
prevents the first secure execution space from executing the same executable accessed by a second secure execution	1. "A second secure execution space having a second tamper resistant barrier with a second **security level** different from the first security level": a second **secure execution space** (different from the first **secure execution space**) is part of the **protected processing environment,** and has a **tamper resistant barrier** (different from the first **tamper resistant barrier**) which has a persistent (not just occasional) **security level** greater or less than the first persistent **security level.**

Claim Phrase	Microsoft's Preliminary Construction
space having a second tamper resistant barrier with a second security level different from the first security level	2. "The same executable accessed by": the same **executable** (as opposed to, e.g., two **copies** of the same **executable**) is simultaneously accessed by both the first **secure execution space** and the second **secure execution space**. 3. "Prevents the first secure execution space from executing": the **arrangement prevents** the first **secure execution space**, otherwise capable of executing the **executable**, from executing any part of the **executable** (e.g., on behalf of any user, process, or device).
'861:58	Claim as a Whole: The recited method is performed within a **VDE**.
creating a first secure container	1. This preamble language is a claim limitation. 2. Completely forming (as opposed to defining) a **secure container** within a **VDE Secure Processing Environment(s)**.
including or addressing . . . organization information . . . desired organization of a content section. . . and metadata information at least in part specifying at least one step required or desired in creation of said first secure container	1. The same single **descriptive data structure** must either **contain** within its confines or **address** both **organization information** and **metadata information**. 2. Both the "desired" organization of the content section and also the "desired" step, occur after the **descriptive data structure** is accessed, not before. 3. The **metadata information** specifies a procedure, as opposed to a result or a **data item**.
at least in part determine specific information required to be included in said first secure container contents	1. The **metadata information** is used to determine the specific value, not merely the kind, of at least some of the information that must be placed inside the **secure container**. 2. The use of the **metadata information** actively requires the **secure container** creation steps to add this specific information to the first **secure container**, as opposed to the specific information being within the **secure container** for some other reason.
rule designed to control at least one aspect of access to or use of at least a portion of said first secure container contents	1. A **rule** designed for these particular **secure container** contents, which is used (by **VDE control(s)** executing in **VDE Secure Processing Environment(s)**) to limit **access** to or **use** of at least a portion of the contents of the first **secure container** (by all users, processes, and devices). Without compliance with this **rule**, no process, user, or device is able to take the **controlled** aspect of the **controlled access** or **use** action.
'891:1	Claim as a Whole: The recited method is performed within a **VDE**.
resource processed in a secure operating	1. This preamble language is a claim limitation. 2. A component part of a first appliance's **secure operating environment** which is processed within that **secure operating environment's** special-

Claim Phrase	Microsoft's Preliminary Construction
environment at a first appliance	purpose **Secure Processing Unit**. A **Secure Processing Unit** is a special-purpose unit isolated from the rest of the world in which a hardware **tamper-resistant barrier encapsulates** a processor and internal **secure memory**. The **barrier** prevents all **unauthorized** interference, removal, observation, and **use** of the information and processes within it. The processor cryptographically verifies the **integrity** of all code loaded from the **secure memory** prior to execution, executes only the code that the processor has **authenticated** for its use, and is otherwise **secure**.
securely receiving a first entity's control at said first appliance	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("securely receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: A first appliance obtaining a **VDE secure container encapsulating** a **control** created, selected, or modified by a first entity, as part of a communication encrypted on the communications level, **authenticating** the first appliance in accordance with **VDE controls associated with** the **secure container**, and accepting the **secure container**.
securely receiving a second entity's control at said first appliance	1. This claim language falls within 35 U.S.C. § 112, ¶ 6. It recites a step or result ("securely receiving") without reciting an action that achieves that result. The specification does not clearly link any particular action to this recited step. Part of the recited function is performed by Communications Controller 666, I/O Controller 600, SPE 503/SPU 500 (particularly "SPU Encryption/Decryption Engine 522" and NVRAM 534b). 2. The recited function requires: A first appliance obtaining a **VDE secure container encapsulating** a **control** created, selected, or modified by a second entity, as part of a communication encrypted on the communications level, **authenticating** the first appliance in accordance with **VDE controls associated with** the **secure container**, and accepting the **secure container**.
securely processing a data item at said first appliance, using at least one resource	1. Performing an operation, inside the special-purpose **Secure Processing Unit** of the first appliance, on a **data item** inside the **Secure Processing Unit**. The operation cannot be observed from outside the **SPU** and is performed only after the **integrity** of the program code for performing such operation is cryptographically verified. A **Secure Processing Unit** is a special-purpose unit isolated from the rest of the world in which a hardware **tamper-resistant barrier encapsulates** a processor and internal **secure memory**. The **barrier** prevents all **unauthorized** interference, removal, observation, and **use** of the information and processes within it. The processor cryptographically verifies the **integrity** of all code loaded from the **secure memory** prior to execution, executes only the code that the processor has **authenticated** for its use, and is otherwise **secure**.
securely applying, at said first appliance through use of said at least one resource said first entity's control and said second entity's control to govern use of said data item	1. Processing the resource (component part of a first appliance's **secure operating environment**) within the **secure operating environment**'s special-purpose **Secure Processing Unit** to execute the first **control** and second **control** in combination within the **SPU**. This execution of these **controls governs** all use of the **data item** by all users, processes, and devices. The processing of the **resource** and execution of the **controls** cannot be observed from outside the **SPU** and is performed only after the **integrity** of the **resource** and **controls** is cryptographically verified. A **Secure Processing Unit** is a special-purpose unit isolated from the rest of the world in which a hardware **tamper-resistant barrier encapsulates** a processor and internal **secure memory**. The **barrier** prevents all **unauthorized** interference, removal, observation, and **use** of the information and processes within it. The processor cryptographically verifies the **integrity** of all code loaded from the **secure memory** prior to execution, executes only the code that the processor has **authenticated** for its use, and is otherwise **secure**.

Claim Phrase	Microsoft's Preliminary Construction
'900:155	Claim as a Whole: The "virtual distribution environment" is **VDE**.
first host processing environment comprising	1. A **host processing environment** that encompasses the recited computer hardware (central processing unit, main **memory**, and mass storage) and certain **VDE Protected Processing Environment** software loaded in that main **memory** and executing in that central processing unit, but does not encompass software, such as the recited **tamper resistant software**, which is stored in mass storage and not executing.
said mass storage storing tamper resistant software	1. The **tamper resistant software** is physically stored within, as opposed to being merely **addressed** by, the mass storage.
designed to be loaded into said main memory and executed by said central processing unit	1. The **tamper resistant software** is capable of being loaded into only said main memory and is capable of being executed only by said central processing unit.
said tamper resistant software comprising: . . . one or more storage locations storing said information	1. The **tamper resistant software** within said mass storage includes one or more storage locations within it. These storage locations are designated to store, and must store, information **derived** by the **machine check programming**, and must not store any other information.
derives information from one or more aspects of said host processing environment,	1. **Deriving** from the **host processing environment** hardware one or more values that uniquely and persistently **identify** the **host processing environment** and distinguish it from other **host processing environments**. 2. The "one or more aspects of said host processing environment" are distinguishing components or parts of the **host processing environment** itself, as opposed to, e.g., data or programs stored within the mass storage or main memory, or processes executing within the **host processing environment**.
one or more storage locations storing said information	1. One or more logical storage locations within the **tamper resistant software** storing only information **derived** by the **machine check programming**.
information previously stored in said one or more storage locations	1. Any information once stored in said "one of more storage locations storing said information," but not stored therein when the recited comparison occurs.
generates an indication based on the result of said comparison	1. Producing an indication based solely on the result of the "compares" step. There are only two possible indications: the comparison found an exact match, or it did not. The "indication" need not be displayed to a user.
programming which takes one or more actions based	1. **Executable programming** code that is a part of the **tamper resistant software**, when executed, and not a part of the **host processing environment**. Whenever the recited indication is generated, no matter what it indicates, this code (executing on the CPU for which it was designed

Claim Phrase	Microsoft's Preliminary Construction
on the state of said indication	and loaded in the **memory** for which it was designed) must take an action, or more than one action. The particular action(s) taken must be based solely on the state of that indication.
at least temporarily halting further processing	1. The action(s) taken by this programming must encompass **halting** or temporarily **halting** all further processing of the **host processing environment** and any processes running within it.
'912:8	Claim as a Whole: The recited method is performed within a **VDE**.
identifying at least one aspect of an execution space required for use and/or execution of the load module	1. Defining fully, without reference to any other information, at least one of the persistent features (aspects) of an **execution space** that are **required** for any use, and/or for any execution, of the **load module**. **An execution space** without all of those **required** aspects is incapable of making any such use (e.g., copying, displaying, printing) and/or execution of the **load module**.
said execution space identifier provides the capability for distinguishing between execution spaces providing a higher level of security and execution spaces providing a lower level of security	1. The **execution space identifier**, by itself, provides the **load module** with the capability of determining the persistent **level of security** of any **execution space** in which it is loaded, and of distinguishing between any two **execution spaces** based on their respective, determined persistent (not just occasional) "**levels of security**." This capability extends to at least two **execution spaces** providing a higher **level of security** and at least two **execution spaces** providing a lower **level of security**.
checking said record for validity prior to performing said executing step	Before executing any **executable programming** encompassed within any element which is directly or indirectly identified by any information **contained** within the first **record**, evaluating, within a **VDE Secure Processing Environment**, the values and formats of all data fields within the first **record** and confirming that they have legitimate values and formats.
'912:35	Claim as a Whole: The recited method is performed within a **VDE**.
received in a secure container	1. The first **processing environment** obtained a **VDE secure container encapsulating** the record inside, and **authenticated** the intended recipient in accordance with **VDE controls associated with** the **secure container**, and accepted the **secure container**.
said component assembly allowing access to or use of specified information	1. The **component assembly** identifies specific information over which it (by itself and with no other information), executing in a **VDE Secure Processing Environment, allows access** or **use** (as opposed to **access** "and" **use**). Unless **allowed** by the **component assembly**, no user, process, or device is able to **access** or **use** the specified information. The **component assembly** is **associated with** and dedicated to this particular specified information.
said first component	1. The first **record** by itself **contains** sufficient information to unambiguously **identify** the assembled **component assembly**, including all of its

Claim Phrase	Microsoft's Preliminary Construction
assembly specified by said first record	elements. 2. This limitation is inconsistent with the recitation "first record containing identification information directly or indirectly identifying one or more elements of first component assembly."

EXHIBIT D

Appended hereto, in accordance with Patent Local Rule 4-2(b), are copies of excerpts of dictionary definitions and other publications.

Exhibit D List of Dictionaries

No.	Dictionary
1	The New IEEE Standard Dictionary of Electrical and Electronic Terms (IEEE 100-1992), 1993, ISBN 1-55937-240-0
2	The Whole Internet: User's Guide and Catalog (O'Reilly & Associates, Inc) ISBN 1-56592-0252
3	Practical Unix Security (O'Reilly & Associates, Inc) ISBN 0-93717-5722
4	Computer Security Basics, Deborah Russell and G.T. Gangemi Sr. (O'Reilly & Associates, 1991) ISBN 0-93717-5714
5	Modern Methods for Computer Security and Privacy, Lance J. Hoffman (Prentice Hall, 1977) ISBN 0-13-595207-7
6	Distributed Systems, Second Edition, Sape Mullender (Addison Wesley, 1993) ISBN 0-20162-4273
7	Formal Models for Computer Security, Carl E Landwehr, ACM Computer Surveys, September 3, 1981 pg 247-275
8	Computer & Communications Security: Strategies for the 1990's, James Arlin Cooper
9	The Computer Security Handbook, Richard Baker (TAB Professional and Reference Books, 1985) ISBN 0-83060-3085
10	Computer Security Handbook 2nd Edition, Hutt, Bosworth, Hoyt (1987) ISBN 002915300X
11	National Information System Security (INFOSEC) Glossary, NSTISSI No. 4009, September 2000
12	Telecommunications: Glossary of Telecommunications Terms by Nation Communications Systems, 1996.
13	Internet Security Glossary, Network Working Group, RFC 2828, May 2000
14	Que's Computer User's Dictionary (1994) ISBN 1-56529-1255
15	The Dictionary of Computing and Digital Media: Terms and Acronyms, Brad Hansen (1999) ISBN 1-887902-38-4
16	Dictionary of Scientific and Technical Terms, 5th ed. (McGraw-Hill, 1994) ISBN 0-07-042333-4
17	The Computer Glossary: The Complete Illustrated Desk Reference, Alan Freedman (Computer Language Co., 1993) ISBN 0-8144-7801-8 (paperback) 0-8144-5104-7 (hardcover)
18	Prentice Hall's Illustrated Dictionary of Computing, 2nd Ed, Jonar C. Nader (Prentice Hall, 1992) ISBN 0-13205-7255
19	Computer Related Risks, Peter G. Neumann (1995) ISBN 0-201-55805-X

20	Dictionary of Computer Science, Engineering and Technology, Phillip A. Laplante (2001) ISBN 0-84932-6915
21	The American Heritage Dictionary of the English Language (1969) Standard Book Reference 0-395-09064-4 or 0-395-09065-2 or 0-395-09066-0
22	Webster's New World Dictionary of Computer Terms (1992) ISBN 0-671-84651-5
23	Webster's College Dictionary of Random House (1991) ISBN 0-679-40110-5 or 0-679-40100-8
24	Dictionary of Computing, Third Edition (Oxford, 1990) ISBN 0-19-853825-1
25	Funk & Wagnalls Standard College Dictionary, 1973
26	Newton's Telecom Dictionary, Harry Newton (1993) ISBN 0-93644-8422; (1996) ISBN 0-93644-8872
27	Tony Gunton, A Dictionary of Information Technology and Computer Science, Second Edition (NCC Blackwell Ltd 1993). ISBN 1-85554-327-3
28	Dictionary of Computer Science, Engineering and Technology, Phillip A. Laplante (2001) ISBN 0-84932-6915
29	Modern Operting Systems, Andrew S. Tanenbaum (Prentice Hall, 1992) ISBN 0-13588-1870
30	Unix System Security, Wood, Kochan (Hayden Books Unix System Library, 1985) ISBN 0-81046-2672
31	Microsoft Computer Dictionary (Microsoft Press, 1994) ISBN 1-55615-597-2
32	Microsoft Computer Dictionary, Third Edition (1997) ISBN 1-57231-446 –X Paperback
33	Security in Computing, Charles P. Pfleeger (Prentice Hall, 1989) 0-13798-9431
34	Information Security: Dictionary of Concepts, Standards and Terms, Dennis Longley, Michael Shain and William Caelli (Stockton Press, 1992) ISBN 1-56159-069-X or 0-333-54698-9
35	The Random House Dictionary of the English Language: College Edition, 1968
36	Dictionary of Object Technology: The Definitive Desk Reference, Donald G Firesmith and Edward M Eykholt (SIGS Book, 1995) ISBN 1-88484-2097
37	Webster's Ninth New Collegiate Dictionary, Merriam-Webster, 1987, ISBN 0-87779-508-8
38	Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe (Benjamin/Dummings Publishing Company, 1989) ISBN 0-80530-1453
39	IBM Dictionary of Computing, George

	McDaniel (McGraw Hill, 1994) ISBN 0-07-031488-8 (hardcover) 0-07031-4896 (paperback)
40	Encyclopedia of Computer Science and Engineering, 2nd Edition (Van Nostrand Reinhold Company, 1983) ISBN 0-4423-24496-7

EXHIBIT E – Production Documents

IN00004247	IN00029902	IN00075912	IN00075923	IN00075949
IN00075983	IN00075989	IN00076751	IN00076879	IN00076896
IN00076920	IN00078052	IN00171981	IN00172167	IN00173457
IN00173475	IN00173653	IN00173934	IN00174227	IN00174246
IN00174289	IN00174303	IN00174357	IN00174373	IN00174376
IN00174435	IN00174623	IN00174656	IN00174689	IN00174702
IN00174718	IN00174728	IN00174745	IN00174748	IN00174751
IN00174763	IN00174796	IN00174828	IN00174851	IN00174895
IN00174925	IN00174945	IN00174975	IN00175067	IN00175077
IN00175093	IN00175103	IN00175115	IN00175137	IN00175145
IN00175286	IN00175300	IN00175323	IN00175354	IN00175602
IN00175621	IN00175709	IN00175713	IN00175729	IN00175843
IN00176182	IN00176184	IN00176253	IN00176256	IN00176259
IN00176261	IN00176315	IN00176319	IN00176373	IN00176417
IN00176432	IN00176437	IN00176453	IN00176542	IN00176868
IN00176901	IN00177015	IN00177409	IN00177425	IN00177442
IN00177450	IN00177490	IN00177501	IN00177524	IN00177539
IN00177542	IN00178045	IN00178050	IN00178098	IN00178157
IN00178502	IN00178643	IN00178686	IN00179081	IN00179430
IN00179571	IN00179772	IN00182386	IN00182542	IN00182584
IN00182604	IN00182621	IN00182642	IN00182678	IN00182685
IN00182732	IN00182741	IN00182801	IN00182844	IN00182852
IN00182863	IN00182879	IN00182902	IN00183008	IN00183129
IN00183177	IN00183254	IN00183325	IN00183379	IN00183391
IN00183523	IN00183660	IN00183698	IN00183699	IN00183711
IN00183904	IN00183912	IN00183916	IN00183952	IN00183973
IN00183983	IN00184010	IN00184020	IN00184024	IN00184038
IN00184113	IN00184121	IN00184149	IN00184376	IN00184498
IN00184663	IN00184819	IN00184858	IN00184864	IN00185251
IN00185254	IN00185263	IN00185319	IN00185347	IN00185361
IN00186679	IN00186694	IN00186707	IN00186711	IN00186750
IN00186761	IN00186890	IN00187009	IN00187458	IN00189107
IN00189456	IN00189583	IN00189815	IN00189818	IN00189828
IN00510231	IN00515938	IN00520960	IN00521077	IN00521959
IN00525310	IN00529007	IN00532191	IN00532292	IN00790333
IN01104824	IN01105073	IN01274376	IN01275382	IN01275395
IN01275439	IN01275511	IT00000005	IT00000067	IT00000207
IT00000230	IT00000243	IT00000259	IT00000280	IT00000293
IT00000315	IT00000343	IT00000371	IT00000399	IT00000401
IT00000406	IT00000715	IT00000964	IT00001419	IT00001510
IT00001515	IT00001522	IT00001533	IT00001569	IT00001650
IT00001663	IT00001678	IT00001689	IT00001698	IT00001711
IT00001729	IT00001746	IT00001753	IT00001762	IT00001784
IT00001820	IT00001851	IT00001876	IT00001897	IT00001935
IT00001953	IT00001960	IT00001968	IT00001994	IT00002002
IT00002030	IT00002079	IT00002090	IT00002097	IT00002101
IT00002103	IT00002133	IT00002155	IT00002160	IT00002163

EXHIBIT E – Production Documents

IT00002178	IT00002183	IT00002186	IT00002202	IT00002228
IT00002263	IT00002279	IT00002285	IT00002324	IT00002326
IT00002346	IT00002607	IT00002657	IT00002716	IT00002731
IT00002742	IT00002774	IT00002881	IT00002887	IT00003151
IT00003184	IT00003361	IT00003378	IT00003394	IT00003438
IT00003439	IT00003443	IT00003453	IT00003462	IT00003792
IT00003803	IT00003851	IT00003871	IT00003874	IT00004021
IT00004136	IT00004555	IT00004891	IT00004973	IT00004979
IT00004980	IT00004993	IT00005067	IT00005068	IT00005090
IT00005102	IT00005262	IT00005306	IT00005314	IT00005401
IT00005416	IT00005450	IT00005463	IT00005749	IT00006228
	IT00006425	IT00006666	IT00006977	IT00007422
IT00007789	IT00007992	IT00008109	IT00008235	IT00008285
IT00008360	IT00008467	IT00008479	IT00008555	IT00009310
IT00009829	IT00009847	IT00009922	IT00010328	IT00010428
IT00010486	IT00010632	IT00010655	IT00010875	IT00011180
IT00012629	IT00012853	IT00012969	IT00013156	IT00013520
IT00013557	IT00013564	IT00013569	IT00013604	IT00013624
IT00013657	IT00013666	IT00013667	IT00013715	IT00013750
IT00013765	IT00014049	IT00014707	IT00014488	IT00014514
IT00014521	IT00014525	IT00014533	IT00014590	IT00014690
IT00014707	IT00014785	IT00014850	IT00014871	IT00014879
IT00014940	IT00015007	IT00015191	IT00015210	IT00015230
IT00015234	IT00015237	IT00015260	IT00015273	IT00015284
IT00015288	IT00015292	IT00015667	IT00015891	IT00016296
IT00016479	IT00016854	IT00018144	IT00018217	IT00018281
IT00018305	IT00018770	IT00018841	IT00018879	IT00021839
IT00025971	IT00026603	IT00026662	IT00026722	IT00026782
IT00026903	IT00026970	IT00027106	IT00027113	IT00027255
IT00027264	IT00027271	IT00027531	IT00027539	IT00027879
IT00027898	IT00027900	IT00027927	IT00027943	IT00027960
IT00027974	IT00027989	IT00028003	IT00028116	IT00028128
IT00028227	IT00028234	IT00028249	IT00028254	IT00028290
IT00028292	IT00028304	IT00028336	IT00028371	IT00028528
IT00028627	IT00028641	IT00028660	IT00030738	IT00031303
IT00031329	IT00031344	IT00031364	IT00031375	IT00031385
IT00031395	IT00031410	IT00031438	IT00031446	IT00031490
IT00031513	IT00031532	IT00031560	IT00031572	IT00031586
IT00031604	IT00031624	IT00031643	IT00031930	IT00031938
IT00031982	IT00032016	IT00032268	IT00032370	IT00032436
IT00036848	IT00036849	IT00036850	IT00036851	IT00036852
IT00036853	IT00036854	IT00036855	IT00036856	IT00036857
IT00036858	IT00036859	IT00036860	IT00036861	IT00036862
IT00036863	IT00036864	IT00036865	IT00036866	IT00036867
IT00036868	IT00036869	IT00036870	IT00036871	IT00036872
IT00036873	IT00036874	IT00036875	IT00036876	IT00036877
IT00036878	IT00037270	IT00037296	IT00037303	IT00037307

EXHIBIT E – Production Documents

IT00037372	IT00037390	IT00037427	IT00037444	IT00037468
IT00037478	IT00037479	IT00037527	IT00037562	IT00037577
IT00038182	IT00038306	IT00038327	IT00038347	IT00038363
IT00038372	IT00038422	IT00038431	IT00038997	IT00039250
IT00039866	IT00040223	IT00040233	IT00040347	IT00040361
IT00040363	IT00041512	IT00041530	IT00042789	IT00045453
IT00045456	IT00045457	IT00045458	IT00046871	IT00049240
IT00050938	IT00050949	IT00050961	IT00051019	IT00051099
IT00051136	IT00051147	IT00051179	IT00051227	IT00051720
IT00051856	IT00051867	IT00051878	IT00051894	IT00051901
IT00051938	IT00051952	IT00052017	IT00052139	IT00052144
IT00052146	IT00052167	IT00052168	IT00054954	IT00055484
IT00069793	IT00095639	IT00095745	IT00095774	IT00095776
IT00095789	IT00095843	IT00095849	IT00095916	IT00095949
IT00096026	IT00096036	IT00096041	IT00096061	IT00096470
IT00096679	IT00096944	IT00097379	IT00097717	IT00113942
IT00114233	IT00114388	IT00114577	IT00114757	IT00117517
IT00117579	IT00117756	IT00117996	IT00118184	IT00118709
IT00118866	IT00119088	IT00119148	IT00119686	IT00119698
IT00119815	IT00119851	IT00119907	IT00119928	IT00120095
IT00120163	IT00120964	IT00121368	IT00121370	IT00121503
IT00122592	IT00124816	IT00124818	IT00124820	IT00124822
IT00124823	IT00124836	IT00124910	IT00124911	IT00124932
IT00124944	IT00125101	IT00125143	IT00125151	IT00125238
IT00125253	IT00125287	IT00132194	IT00132319	IT00132467
IT00132502	IT00132505	IT00132554	IT00132864	IT00133375
IT00133709	IT00134029	IT00134508	IT00135017	IT00135382
IT00136266	IT00136488	IT00136548	IT00137267	IT00138348
IT00138641	IT00139967	IT00140269	IT00140401	IT00141322
IT00142337	IT00143975	IT00149724	IT00150856	IT00167103
IT00169208	IT00177126	IT00188324	IT00191480	IT00192334
IT00197477	IT00197482	IT00197491	IT00210571	IT00214347
IT00247624	IT00340503	IT00383482	IT00399871	IT00399877
IT00421590	IT00516589	IT00523579	IT00523595	IT00523612
IT00523646	IT00523662	IT00523677	IT00523696	IT00523701
IT00543741	IT00550662	IT00559320	IT00560258	IT00565857
IT00566022	IT00698363	IT00698378	IT00698393	IT00698409
IT00702804	IT00702823	IT00702855	IT00702862	IT00702864
IT00702869	IT00702891	IT00702908	IT00702922	IT00702937
IT00702969	IT00702987	IT00703050	IT00703055	IT00703080
IT00703081	IT00703083	IT00703090	IT00703100	IT00703152
IT00703217	IT00703258	IT00703287	IT00703326	IT00703390
IT00703400	IT00703419	IT00703435	IT00703477	460MIC00007
460MIC00017	GNT00000001	GNT00004298	GNT00004304	GNT00004444
GNT00004851	GNT00004927	GNT00004958	LINN00001-709	NIST00001-2226
NSA 00275-3033	NV00000037	WW00143		

EXHIBIT F - Publications

MSI0156840-45
MSI017361-62
MSI017548-556
MSI017864-70
MSI017899-911
MSI018529-74
MSI018811-19
MSI019082-99
MSI020348-370
MSI020373-384
MSI020389-396
MSI021744
MSI022371-380
MSI022440-455
MSI022766-23028
MSI027045-068
MSI036052-92
MSI036113-28
MSI067984-8007
MSI068022 et seq.
MSI079611-656
MSI079796-806
MSI079844-49
MSI079850-56
MSI079850-56
MSI080030-40
MSI080041-42
MSI080043-97
MSI080098-102
MSI080103-116
MSI080117-119
MSI080120-31
MSI080160-164
MSI080173-188
MSI080194-204
MSI080205-222
MSI080254-260
MSI080261-276
MSI080290-302
MSI080337-348
MSI080349-55
MSI080356-397
MSI080410-419
MSI080420-422
MSI080441-445
MSI080456-478
MSI080562-574
MSI080575-696
MSI080697-712
MSI080779-784
MSI080875-909
MSI080910-26
MSI080927-973
MSI080974-81007
MSI081052-66
MSI081140 et seq
MSI081240 et seq.
MSI081464 et seq.
MSI081464-717
MSI082792-826
MSI082958-3002
MSI083003-13
MSI083105-107
MSI083108-179
MSI083356 et seq.
MSI083356-99
MSI083400
MSI083410
MSI083423
MSI083423-443
MSI083444 et seq.
MSI085035-6
MSI085043-48
MSI085049-58
MSI085078-114
MSI085115-35
MSI085136-149
MSI085189-200
MSI085211
MSI085217-29
MSI085230-44
MSI085245-350
MSI085479-521
MSI085551-68
MSI085569
MSI085569-92
MSI085593-653
MSI085654
MSI085704 et seq.
MSI085756
MSI085831-33
MSI085834-36
MSI085837-39
MSI085840-42
MSI085843-44
MSI086146-55
MSI086641-52
MSI086653-62
MSI086675-86
MSI086687-703
MSI086704-727
MSI086845-863
MSI086864-67
MSI086893-904
MSI086905-919
MSI086923-25
MSI086926-32
MSI086946-950
MSI086951-73
MSI086985-87006
MSI087007-36
MSI087007-36
MSI087037-43
MSI087044-46
MSI087047-60
MSI087061-80
MSI087081-88
MSI087089-106
MSI087107-14
MSI087153-54
MSI087160-62
MSI087341-51
MSI087352-364
MSI087365-86
MSI087392-407
MSI087408-421
MSI087422-443
MSI087444-447
MSI087448-453
MSI087454-465
MSI087519-31
MSI087532-45

EXHIBIT F - Publications

MSI087558-75
MSI087576-85
MSI087586-592
MSI087598-623
MSI087681-693
MSI087694-707
MSI087717-24
MSI087725-36
MSI087737-47
MSI087748-64
MSI087765-75
MSI087776-793
MSI087811-820
MSI087821-838
MSI087839-51
MSI087852-65
MSI088177-79
MSI088186-203
MSI088297
MSI088297-313
MSI088314
MSI088342
MSI088363
MSI088368
MSI088508512
MSI088562-759
MSI088773-80
MSI088781-95
MSI088820-73
MSI088876-78
MSI088876-78
MSI088879-80
MSI088881-83
MSI088884-85
MSI088886-87
MSI088888-89
MSI088890-91
MSI088892-94
MSI088896-902
MSI088903-13
MSI088914-18
MSI088919-33
MSI088938-9026
MSI088953
MSI089068-165
MSI089191-200
MSI089974-90004
MSI090025-45
MSI090055-66
MSI090091-114
MSI090181-244
MSI093870-94277
MSI094278
MSI094738 et seq.
MSI095044 et seq.
MSI095787 et seq.
MSI096004 et seq.
MSI120670-720
MSI132777-87
MSI156788-93
MSI156834-39
MSI161724-31
MSI161742-52
MSI161753-60
MSI161761-96
MSI161797-800
MSI161856-65
MSI161886-94
MSI161895-901
MSI161921-42
MSI161943-68
MSI161969-85
MSI162040-60
MSI162061-75
MSI162132 et seq.
MSI162160-355
MSI162646
MSI162654-69
MSI162698-710
MSI162729-45
MSI162756-76
MSI162788-800
MSI162820-23
MSI162830-37
MSI162841-45
MSI163014-79
MSI163338-39
MSI163355-57
MSI163358-59
MSI163519-23
MSI163935-4204
MSI164205-4281
MSI164771-800
MSI164842-46
MSI168115-131
MSI168132-56
MSI168350-385
MSI173839-868
MSI173884-886
MSI173887-892
MSI173893-904
MSI173905-906
MSI175647-669
MSI175666-704
MSI175722-730
MSI17573-739
MSI175738-763
MSI175740-748
MSI175764-787
MSI175788-795
MSI175818-839
MSI175840-903
MSI175869-934
MSI175904-911
MSI175912-944
MSI175935-943
MSI175944-978
MSI176004-070
MSI176033-073
MSI176183-204
MSI192494-509
MSI192652-60
MSI192773-87
MSI197506-8536
MSI198537-8578
MSI198579-8690
MSI198691-8846
MSI199341
MSI199404
MSI199417
MSI199425
MSI199480

EXHIBIT F - Publications

MSI199485
MSI199525
MSI199536
MSI199550
MSI199576
MSI199605
MSI199612
MSI199618
MSI199633
MSI199645
MSI199681
MSI199700
MSI199736
MSI199738
MSI199743
MSI199755
MSI204966
MSI204983
MSI205081
MSI205118
MSI205137
MSI205190
MSI205529
MSI205548
MSI205621
MSI205633
MSI205648
MSI205650
MSI205659
MSI205660
MSI205749
MSI205967
MSI205981
MSI206028
MSI206144
MSI206178
MSI206672
MSI206764-76
MSI206777
MSI206951
MSI206952
MSI206982
MSI207089
MSI207745-80
MSI207781-819
MSI207820-46
MSI207847-72
MSI207873-901
MSI207998-8022
MSI208023-45
MSI208222-79
MSI208280-320
MSI208321-63
MSI208364-98
MSI208434-64
MSI208465-501
MSI208843-67
MSI209487-95
MSI209496-524
MSI209525-46
MSI209547-58
MSI209692-712
MSI209713-26
MSI209854-68
MSI209981-90
MSI210379-402
MSI210403-20
MSI210421-27
MSI210447
MSI210496-516
MSI210517-21
MSI210522-30
MSI210838-55
MSI210856-899
MSI210900-40
MSI210941-83
MSI210984-90
MSI211023-24
MSI211026-27
MSI211071-132
MSI211133-79
MSI211180-324
MSI211325-465
MSI211504-5
MSI211515-17
MSI211582-610
MSI213017-25
MSI213026-41
MSI213042-65
MSI213076-87
MSI213088-96
MSI213206-10
MSI213211-13
MSI213214-20

(see also cited production documents such as LINN, NIST)

Exhibit F Publications

Bates Number	Author and Title
MSI0156840	Willett, S., "Metered PCs: Is Your System Watching You?, Wave Systems Beta Tests New Technology," May 2, 1994.
MSI017361	Cox, B., "Superdistribution," January 1996.
MSI017548	**Hauser, Ralf, C., "Does Licensing Require New Access Control Techniques?," August 12, 1993.**
MSI017864	Purdy, G.B., et al., "A Software Protection Scheme," April 1982.
MSI017899	**Sibert, O., et al., "The DigiBox: A Self-Protecting Container for Information Commerce," July 1995.**
MSI018529	**Lampson, B., et al. "Authentication in Distributed Systems in Theory and Practice," November 4, 1992.**
MSI018811	**Newman, B.C., "Proxy-Based Authorization and Accounting for Distributed Systems," 1993.**
MSI019082	CUPID, "Protocols and Services (Version 1): An Architectural Overview," November 1997.
MSI020348	Herzberg, A., et al., "Public Protection of Software," November 1987.
MSI020373	Hot Java: "The Security Story," 1996
MSI020389	JavaSoft, "Frequently Asked Questions-Applet Security," June 7, 1996.
MSI021744	Arneke, D., "AT&T Encryption System Protects Information Services" News Release, January 9, 1995.
MSI022371	Low, S.H., et al., "Anonymous Credit Cards," November 1994.
MSI022440	Kristol, D.M., et al., "Anonymous Internet Mercentile Protocol," March 17, 1994.
MSI022766	**The First USENIX Workshop on Electronic Commerce Proceedings, July 1995.**
MSI027045	**Denning, Dorothy E., et al., "Data Security," September 1979.**
MSI036052	**Wobber, E., et al. "Authenticating in the Taos Operating System," Digital Systems Research Center," December 10, 1993.**
MSI036113	**Olson, M., et al., "Concurrent Access Licensing," 1988.**

Exhibit F Publications

MSI067984	Kahn, D., "The Codebreakers: The Story of Secret Writing," December 1996.
MSI068022	**Cohen, F.B., "Protection and Security on the Information Superhighway," 1995.**
MSI079611	**Brockschmidt, Kraig, "Inside OLE 2," 1994.**
MSI079796	Ioannidis, J., et al., "The Architecture and Implementation of Network-Layer Security Under Unix," October 1993.
MSI079844	Berners-Lee, T.J., et al., "Networked Information Services: The World-Wide Web," 1992.
MSI079850	Rivest, R.L., et al., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," 1978.
MSI079850	Rivest, R.L., et al., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," 1978.
MSI080030	Ferraiolo, D., et al., "Role-Based Access Control," 1992.
MSI080041	NIST, "Role Based Access Control," 1992.
MSI080043	U.S. Department of Commerce (NIST), "Security Requirements for Cryptographic Modules," undated.
MSI080098	Kreutzer, A.N., "An On-Line System for Controlling and Monitoring Software Usage in a Microcomputer Laboratory," August 1988.
MSI080103	**Millen, J.K., et al., "Security for Object-Oriented Database Systems," May 1992.**
MSI080117	Kim, L., et al., "Novell Cuisine," 1991.
MSI080120	**Koenig, A., "Automatic Software Distribution," June 1984.**
MSI080160	**Leiss, E., "On Authorization Systems with Grantor-Controlled Propagation on Privileges," 1983.**
MSI080173	**Strack, Hermann, "Extended Access Control in UNIX System V-ACLs and Context," August 1990.**
MSI080194	**Polk, T.W., "Approximating Clark-Wilson "Access Triples" with Basic UNIX Controls," October 1993.**
MSI080205	Kamens, J.I., "Retrofitting Network Security to Third-Party Applications-The SecureBase Experience," October 1993.

Exhibit F Publications

MSI080254	**Gamble, Todd, "Implementing Execution Controls in Unix," 1993.**
MSI080261	**Epstein, J., "A Trusted X Window System Server for Trusted Mach," August 30, 1990.**
MSI080290	Kelter, U., "Discretionary Access Controls in a High-Performance Object Management System," 1991.
MSI080337	Downs, D.D., et al., "Issues in Discretionary Access Control," April 1985.
MSI080349	**Solomon, Daniel, J., "Processing Multilevel Secure Objects," April 1981.**
MSI080356	**Kim, Won, et al., "Object-Oriented Concepts, Databases, and Applications," 1989.**
MSI080410	Maude, T., et al., "Hardware Protection Against Software Piracy," September 1984.
MSI080420	Lipson, S., "Little Black Box 'Blocks' Illicit Software Copying," September 14, 1986.
MSI080441	**Sandhu, Ravi, S., et al., "Implementation Considerations for the Typed Access Matrix Model in a Distributed Environment," October 1992.**
MSI080456	Rescorla, E., et al., "The Secure HyperText Transfer Protocol," June 1994.
MSI080562	**Keefe, T.F., et al., "Prototyping the SODA Security Model," 1990.**
MSI080575	**National Security Agency, "A Guide to Understanding Security Modeling in Trusted Systems," October 1992.**
MSI080697	Sandhu, R.S., "The Typed Access Matrix Model," May 1992.
MSI080779	**Lunt, Teresa, "Multilevel Security for Object-Oriented Database Systems," 1990.**
MSI080875	Tygar, J.D., et al., "Strongbox: A System for Self-Securing Programs," 1991.
MSI080910	**Yee, B., et al., "Secure Coprocessors in Electronic Commerce Applications," July 1995.**
MSI080927	**Franklin, M., et al., "An Overview of Secure Distribution Computing," March 24, 1992.**
MSI080974	Anderson, R., "Why Cryptosystems Fail," November 1993.
MSI081052	**Bishop, M., "Anatomy of a Proactive Password Changer," 1992.**

Exhibit F Publications

MSI081140	**Time-Life Books, "Understanding Computers: Computer Security," 1986.**
MSI081240	**Wood, P.H., et al., "UNIX System Security," 1985.**
MSI081464	**Custer, Helen, "Inside Windows NT," 1993.**
MSI081464	**Custer, Helen, "Inside Windows NT," 1993.**
MSI082792	Symantec Corporation, "THINK Pascal: The Fastest Way to Finished Software," 1990.
MSI082958	**Olivier, M.S., et al., "A Taxonomy for Secure Object-Oriented Databases," 1994.**
MSI083003	**McCollum, C.J., et al., "Beyond the Pale of MAC and DAC-Defining New Forms of Access Control," 1990.**
MSI083105	"Mach Books," viewed on February 6, 2002 at <http://www2.cs.cmu.edu/afs/cs/project/mach/public/www/doc/books.html.>
MSI083108	**Department of Defense, "Trusted Computer System Evaluation Criteria," Dece,ber 1985.**
MSI083356	**LaLonde Wilf, R., et al., "Inside Smalltalk," Volume 1, 1990.**
MSI083356	**LaLonde Wilf, R., et al., "Inside Smalltalk," Volume 1, 1990.**
MSI083400	**Muftic, Sead, "Security Mechanisms for Computer Networks," 1989.**
MSI083410	**Konheim, Alan, G., et al.,"Cryptography: A Primer," 1981.**
MSI083423	**Davies, D.W., et al., "Security for Computer Networks," 1984.**
MSI083423	**Davies, D.W., et al., "Security for Computer Networks," 1984.**
MSI083444	**Castano, S., et al., "Database Security," 1995.**
MSI085035	**OOPSLA 1993: Addendum to the Proceedings, "Security for Object-Oriented Systems," September 26-October 1, 1993.**
MSI085043	Thuraisingham, B., et al., "Parallel Processing and Trusted Database Management Systems," 1993.
MSI085049	Secure Computing, "Constructing a High Assurance Mail Guard," 1994.
MSI085078	Sandhu, R.S., et al., "Data and Database Security and Controls," 1993.

Exhibit F Publications

MSI085115	Thomas, R.K., et al., "Implementing the Message Filter Object-Oriented Security Model without Trusted Subjects," August 1992.
MSI085136	Sandhu, R., et al., "A Secure Kernelized Architecture for Multilevel Object-Oriented Databases," June 1991.
MSI085189	Sibert, O., et al., "The Intel 80x86 Processor Architecture: Pitfalls for Secure Systems," May 1995.
MSI085211	Ware, W., Chairman RAND Corporation "Panel: The InterTrust Commerce Architecture," 1997.
MSI085217	**Fine, T.,et al, "Assuring Distributed Trusted Mach," 1993.**
MSI085230	Minear, S.E., "Providing Policy Control Over Object Operations in a Mach Based System," April 28, 1995.
MSI085245	**Harris, J., et al., "Bento Specification," July 15, 1993.**
MSI085479	Young, W.D., "Verifiable Computer Security and Hardware: Issues," September 1991.
MSI085551	Denning, A., "OLE Controls Inside Out," 1995.
MSI085569	**Denning, D., "Crytography and Data Security," 1982.**
MSI085569	**Denning, D., "Crytography and Data Security," 1982.**
MSI085593	**Schneier, Bruce, "Applied Cryptography: Protocols, Algorithms, and Source Code in C," 1994.**
MSI085654	**Garfinkel, Simson, et al., "Practical UNIX Security," 1991.**
MSI085704	**Dougherty, D., et al., "The Mosaic Handbook for the X Window System," 1994.**
MSI085756	**Curry, D.A., "UNIX System Security: A Guide for Users and System Administrators," 1992.**
MSI085831	International Infrastructure Standards Panel, "IISP Need #31-Containers or Secure Packaging," September 18, 1995.
MSI085834	International Infrastructure Standards Panel, "IISP Need #32-Authentication of Content," September 18, 1995.
MSI085837	International Infrastructure Standards Panel, "IISP Need #33-Control Enforcement," September 18, 1995.

Exhibit F Publications

MSI085840	International Infrastructure Standards Panel, "IISP Need #34-Billing and Payment," September 18, 1995
MSI085843	International Infrastructure Standards Panel, "IISP Need #35-Reporting," September 18, 1995.
MSI086146	Kaplan, M., "IBM Cryptolopes, Super Distribution and Digital Rights Management," December 30, 1996.
MSI086641	Sebes, E.J., et al., "The Triad System: The Design of a Distributed, Real-Time, Trusted System," January 22, 1998.
MSI086653	Sebes, E.J., et al., "The Architecture of Triad: A Distributed, Real-Time, Trusted System," (undated).
MSI086675	Sebes, E.J., "Overview of the Architecture of Distributed Trusted Mach," undated.
MSI086687	Vickers Benzel, T.C., et al., "Identification of Subjects and Objects in a Trusted Extensible Client Server Architecture," undated.
MSI086704	Borenstein, N., "MIME Extensions for Mail-Enabled Applications: Application/Safe-Tel and Multipart/Enabled-Mail," November 1993.
MSI086845	Bellare, M., "iKP-A Family of Secure Electronic Payment Protocols," April 16, 1995.
MSI086864	**Harn, Lein, et al., "A Software Authentication System for the Prevention of Computer Viruses," 1992.**
MSI086893	Deutsch, P., "GZIP File Format Specification Version 4.3," May 1996.
MSI086905	**Gong, Li, "A Secure Identity-Based Capability System," January 1989.**
MSI086923	Merkle, R.C., "Secure Communications Over Insecure Channels," April 1978.
MSI086926	Denning, D.E., "Secure Personal Computing in an Insecure Network," August 1979.
MSI086946	**Medvinsky, G., et al., "NetCash: A Design for Practical Electronic Currency on the Internet," 1993.**
MSI086951	Franz, M., "Technological Steps Toward a Software Component Industry," undated.
MSI086985	Kent, S., et al., "Privacy Enhancement for Internet Electronic Mail: Part II--Certificate-Based Key Management," August 1989.

Exhibit F Publications

MSI087007	Linn, J., "Privacy Enhancement for Internet Electronic Mail: Part I--Message Encipherment and Authentication Procedures," August 1989.
MSI087007	Linn, J., "Privacy Enhancement for Internet Electronic Mail: Part I--Message Encipherment and Authentication Procedures," August 1989.
MSI087037	**Chaum, D., "Achieving Electronic Privacy," August 1992.**
MSI087044	Microsoft Press, OLE 2 Programmer's Reference; Volume 1, "Working with Windows Objects," 1994.
MSI087047	**Brockschmidt, K., "A Primer on Designing Custom Controls," March/April 1992.**
MSI087061	Klemond, P., "Taking the Bull by the Horns: Investigating Object Linking and Embedding, Part I," March/April 1992.
MSI087081	Klemond, P., "Investigating Object Linking and Embedding, Part II: Adding Server Support," May/June 1992.
MSI087089	**DiLascia, Paul, "OLE Made Almost Easy: Creating Containers and Servers Using MFC 2.5," April 1994.**
MSI087107	**Coad, Peter, "Object-Oriented Patterns," September 1992.**
MSI087153	Zelnick, Nate, "Keeping Business Safe on the Internet," April 25, 1995.
MSI087160	**Lucent Technologies, "AT&T Encryption System Protects Information Services," January 9, 1995.**
MSI087341	**Birrell, Andrew, D., et al., "A Global Authentication Service Without Global Trust," April 1986.**
MSI087352	Gasser, M., et al., "The Digital Distributed System Security Architecture," 1989.
MSI087365	**Abadi, M., et al., "Authentication and Delegation with Smart-Cards," 1990.**
MSI087392	Zeleznick, M.P., "Security Design in Distributed Computing Applications," December 1993.
MSI087408	**Moffett, Jonathan, D., et al., "An Introduction to Security Distributed Systems," August 1993.**
MSI087422	Netscape, "SSL 2.0 Protocol Specification," February 9, 1995.

Exhibit F Publications

MSI087444	Davis, D., et al., "Network Security via Private-Key Certificates," October 1990.
MSI087448	Gruber, R., et al., "Disconnected Operation in the Thor Object-Oriented Database System," December 1994.
MSI087454	**Ting, T.C., et al., "Requirements, Capabilities and Functionalities of User Role Based Security for an Object-Oriented Design Model," 1992.**
MSI087519	**Gifford, David, K., "Crytographic Sealing for Information Secrecy and Authentication," 1982.**
MSI087532	Boly, J.P., et al., "The ESPIRIT Project CAFÉ: High Security Digital Payment Systems," 1994.
MSI087558	**Kluepfel, H.M., "Securing a Global Village and its Resources: Baseline Security for Interconnected Signaling System #7 Telecommunications Networks," 1993.**
MSI087576	**Uhler, Stephen A., "PhoneStation, Moving the Telephone onto the Virtual Desktop," January 1993.**
MSI087586	**Brown, Patrick W., "Digital Signatures: Can They Be Accepted as Legal Signatures in EDI?," November 1993.**
MSI087598	**Popek, Gerald, J., et al., "Encryption and Secure Computer Networks," December 1979.**
MSI087681	Picciotto, J., et al., "Extended Labeling Policies for Enhanced Application Support," 1994.
MSI087694	Born, E, et al., "Discretionary Access Control by Means of Usage Conditions," 1994.
MSI087717	Russell, S., "Paradigms for Verification of Authorization at Source of Electronic Documents in an Integrated Environment," 1993.
MSI087725	**Olivier, M.S., et al., "Building a Secure Database Using Self-Protecting Objects," 1992.**
MSI087737	**Press, J., "Secure Transfer of Identity and Privilege Attributes in an Open Systems Environment," 1991.**
MSI087748	**Fugini, M.G., et al., "Authorization and Access Control in the Office-Net System," 1989.**
MSI087765	**Hardjono, Thomas, "Record Encryption in Distributed Databases," January 1990.**

Exhibit F Publications

MSI087776	Calas, C., "Distributed File System Over a Multilevel Secure Architecture Problems and Solutions," 1994.
MSI087811	**Russell, S., "Planning for the EDI of Tomorrow Using Electronic Document Authorization," 1993.**
MSI087821	Lin, P., "The Encapsulated Security Services Interface (ESSI)," 1993.
MSI087839	**Fugini, M., et al., "Security Management in Office Information Systems," 1984.**
MSI087852	Montini, G, et al., "Access Control Models and Office Structures," 1984.
MSI088177	Multics, Home; viewed on November 12, 2001 at <http://www.multicians.org>.
MSI088186	Corbato, F.J., et al., "Introduction and Overview of the Multics System," 1992.
MSI088297	Moffett, J., et al., "Specifying Discretionary Access Control Policy for Distributed Systems," September 18, 1990.
MSI088297	Moffett, J., et al., "Specifying Discretionary Access Control Policy for Distributed Systems," September 18, 1990.
MSI088314	Moffett, J.D., "Specification of Management Policies and Discretionary Access Control," June 28, 1994.
MSI088342	Moffett, J.D., "Specification of Management Policies and Discretionary Access Control," 1994.
MSI088363	Moffett, J.D.,et al., "Policy Hierarchies for Distributed Systems Management," December 1993.
MSI088368	Moffett, J.D., et al., "The Representation of Policies as System Objects," November 1991.
MSI088508	Lampson, B.W., "A Note on the Confinement Problem," 1973.
MSI088562	**Olivier, M.S., "Secure Object-Oriented Databases," December 1991.**
MSI088773	Thuraisingham, M.B., "Mandatory Security in Object-Oriented Database Systems," October 1989.
MSI088781	**Chaum, D., "Security Without Identification: Transaction Systems to Make Big Brother Obsolete," October 1985.**
MSI088820	**Lampson, B., "Computer Security," 1991.**

Exhibit F Publications

MSI088876	Walker, S., "Notes from RSA Data Security Conference," January 18, 1994.
MSI088876	Walker, S., "Notes from RSA Data Security Conference," January 18, 1994.
MSI088879	RSA Security; News; <http://rsasecurity.com/news/pr/9401.html.>, dated January 12, 1994.
MSI088881	RSA Security; News; "iPower's Data Security Approach," <http://rsasecurity.com/news/pr/940112-10.html.>, dated January 12, 1994.
MSI088884	RSA Security; News; "General Magic Picks RSA," <http://rsasecurity.com/news/pr/940112-3.html.>, dated January 12, 1994.
MSI088886	RSA Security; News; "Enterprise Solutions Announces RSA Mail," <http://rsasecurity.com/news/pr/940112-2.html.>, dated January 12, 1994.
MSI088888	RSA Security; News; "Hewlett-Packard Chooses RSA," <http://rsasecurity.com/news/pr/940112-5.html.>, dated January 12, 1994.
MSI088890	RSA Security; News; "Hilgraeve Ships Secure Version of HyperACCESS/5," <http://rsasecurity.com/news/pr/940112-8.html.>, dated January 12, 1994.
MSI088892	RSA Security; News; "RSA Enters Wireless Arena," <http://rsasecurity.com/news/pr/940112-6.html.>, dated January 12, 1994.
MSI088896	Williams, S., "An MSJ Interview with Microsoft's Chief Architect of OLE, Tony Williams," October 1993.
MSI088903	Brockschmidt, K., "OLE 2.0 Part II: Implementing a Simple Windows Object Using Either C or C++," October 1993.
MSI088914	Brockschmidt, K., "Introducing OLE 2.0, Part 1: Windows Objects and the Component Object Model," August 1993.
MSI088919	Brockschmidt, K., "Implementing OLE 2.0, Part III: Uniform Data Transfer with Data Objects," December 1993.
MSI088938	Townsend, J.E., "NIST on Internet Security," March 22, 1994.
MSI088953	Curry, David A., "Improving the Security of Your Unix System," April 1990.
MSI089068	Kohl, J., "The Kerberos Network Authentication Services (V5)," September 1993.

Exhibit F Publications

MSI089191	Bertino, Elisa, "Data Hiding and Security in Object-Oriented Databases," 1992.
MSI089974	Kim, W., et al., "Features of the ORION Object-Oriented Database System," 1989.
MSI090025	Kelter, U., et al., "Type Level Access Controls for Distributed Structurally Object-Oriented Database Systems," November 1992.
MSI090055	Wong, R., et al., "The SIDOS System: A Secure Distributed Operating System Prototype," October 1989.
MSI090091	OMG Security Working Group, "OMG White Paper on Security," April 1994.
MSI090181	National Computer Security Center, "Trusted Unix Working Group (TRUSIX) Rationale for Selecting Access Control List Features for the UNIX (R) System," August 18, 1989.
MSI093870	UNKNOWN
MSI094278	**UNKNOWN**
MSI094738	**UNKNOWN**
MSI095044	Tanenbaum, A.S., "Operating Systems: Design and Implementation," 1987.
MSI095787	Date, C.J., "An Introduction to Database Systems," 4th. Ed., Vol. 1, 1987.
MSI096004	Tanenbaum, A.S., "Modern Operating Systems," 1992.
MSI120670	UNKNOWN
MSI132777	E-mail from Caglar Gunyakti entitled: "Private Test Needed," April 28, 2001.
MSI156788	**Clark, Paul, C., et al., "BITS: A Smartcard Protected Operating System," November 1994.**
MSI156834	**O'Connor, MaryAnn, "New Distribution Option for Electronic Publishers," March 1994.**
MSI161724	**Blaze, Matt, "A Crytographic File System for Unix," November 1993.**
MSI161742	Ferraiolo, D., et al., "Role-Based Access Control," 1992.
MSI161753	Hardy, N., "The Keykos Architecture," December 1990.
MSI161761	Miller, S.P., et al., "Kerberos Authentication and Authorization System," October 27, 1998.

Exhibit F Publications

	October 27, 1998.
MSI161797	IBM, "OpenDoc vs. OLE 2.0: Superior by Design," January 1994.
MSI161856	Brickell, E.F., et al., "The SKIPJACK Algorithm," July 28, 1993.
MSI161886	Blaze, M., "Key Management in an Encrypting File System," 1994.
MSI161895	**Woo, Thomas, Y.C., et al., "A Framework for Distributed Authorization," November 1993.**
MSI161921	Davin, J., et al., "SNMP Administrative Model," July 1992.
MSI161943	**Galvin, J., et al., "SNMP Security Protocols," July 1992.**
MSI161969	McCloghrie, K., et al., "Definitions of Managed Objects for Administration of SNMP Parties," July 1992.
MSI162040	Case, J., "A Simple Network Management Protocol (SNMP)," May 1990.
MSI162061	Information Systems Audit and Control Association-Montreal Chaper, "Authentification dans les environnements de traitement distributes," undated.
MSI162132	**Meyer, C.H., et al., "Cryptography: A New Dimension in Computer Data Security," 1982.**
MSI162160	Hewlett Packard Co., "Manager's Guide to MPE/iX Security," April 1994.
MSI162646	Hansen, S.E., et al., "Automated System Monitoring and Notification with Swatch," November 1993.
MSI162654	Bellovin, S.M., "There Be Dragons," August 15, 1992.
MSI162698	Bellovin, S.M., "Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks," May 1992.
MSI162729	Brands, S., "Untraceable Off-line Cash in Wallets with Observers," 1993.
MSI162756	White, J.E., "Telescript: The Foundation for the Electronic Marketplace," November 30, 1993.
MSI162788	White, J.E., "Telescript Technology: An Introduction to the Language," 1994.
MSI162820	Johnson, R., "Info on Telescript," December 6, 1994.

Exhibit F Publications

MSI162830	"An Introduction to Safety and Security in Telescript," undated.
MSI162841	Gosling, J., "Oak Intermediate Bytecodes," 1995.
MSI163014	Kohl, J., et al, "The Kerberos Network Authentication Service," June 30, 1991.
MSI163338	"A Brief History of the Green Project," viewed on March 12, 2002 at <http://java.sun.com/people/jag/green/index.html>.
MSI163355	IBM,"IBM Cryptolope Technology-Executive Summary," viewed on March 13, 2002 at <http://www-3.ibm.com/software/security/cryptolope.about.html>.
MSI163358	IBM, "Key Cryptolope Components," viewed on March 13, 2002 at <http://www-3.ibm.com/software/security/cryptolope.about.html>.
MSI163519	Stubblebine, S.G., "Security Services for Multimedia Conferencing," September , 1993.
MSI163935	NIST & NSA, "Federal Criteria Information Technology Security," Vol. II, Version 1.0, December 1992.
MSI164205	Rashid, R.F., "CMU Computer Science: A 25th Anniversary Commemorative," 1991.
MSI164771	**Multics History, visited on October 8, 1991 at <http://www.multicians.org/history.html>.**
MSI164842	**Halfhill, Tom, R., et al., "Just Like Magic?," February 1994.**
MSI168115	Chaum, D., et al., "Wallet Databases with Observers," 1998.
MSI168132	**Walker, Bruce, J., et al., "Computer Security and Protection Structures," 1977.**
MSI168350	Stallings, W., "Cryptography and Network Security: Principles and Practice," 1999.
MSI173839	Levine, P.H., et al., "Apollo Network License Server," October 1987.
MSI173884	Cabell, D., et al., "Software Protection," May 1985.
MSI173887	Glatzer, H., "The Promise of LANs MIS Back in Control," March 1985.
MSI173893	Cook, S., "Net Results," December 1985.
MSI173905	Kramer, M., "Strength in Numbers," July 22, 1986.

Exhibit F Publications

MSI175647	Custer, H., "Inside the Windows NT File System," 1994.
MSI175666	Appendix for Custer, H., "Inside the Windows NT File System," 1994.
MSI175722	Index for Custer, H., "Inside the Windows NT File System," 1994.
MSI17573	Solomon, A., "PC Viruses: Detection, Analysis and Cure," November 1991.
MSI175738	Glossary for Solomon, A., "PC Viruses: Detection, Analysis and Cure," November 1991.
MSI175740	Roberts, R., et al., Compute!'s "Computer Security," 1989.
MSI175764	Bibliography for Shaffer, S.L., et al., "Network Security," 1994.
MSI175788	Contents for Denning, P.J., "Computers Under Attack; Intruders, Worms & Viruses," 1990.
MSI175818	Keyword Index for Horster, P., "Communications and Multimedia Security II," 1996.
MSI175840	Index for Park, J.S., "AS/400 Security in a Client/Server Environment," 1995.
MSI175869	Diffie, et al., "Privacy on the Line: The Politics of Wiretapping and Encryption," 1998.
MSI175904	Bibliography for Diffie, et al., "Privacy on the Line: The Politics of Wiretapping and Encryption," 1998.
MSI175912	Index for Diffie, et al., "Privacy on the Line: The Politics of Wiretapping and Encryption," 1998.
MSI175935	Baker, R.H., "The Computer Security Handbook," 1985.
MSI175944	Hoffman, L.J., "Modern Methods for Computer Security and Privacy," 1977.
MSI176004	Krol, E., "The Whole Internet User's Guide and Catalog," 2nd. Ed., 1992.
MSI176033	Glossary for Krol, E., "The Whole Internet User's Guide and Catalog," 2nd. Ed., 1992.
MSI176183	Holsinger, E., "How Music and Computers Work," 1994.
MSI192494	**Bell-Labs Secure Technologies, "Information Vending Encryption System (IVES)™," May 31, 2002.**
MSI192652	Rubin, A.D., "Trusted Distribution of Software Over the Internet," 1995.

Exhibit F Publications

MSI192773	Kohl, J.T., et al., "The Evolution of the Kerberos Authentication Service," undated.
MSI197506	CardTech/SecurTech 94 Conference Proceedings, "Building Foundations for Innovation," April 1994.
MSI198537	Smart Card 1993 Conference Proceedings, "Day 1: Communications and Marketing Systems & Market Overview," 1993.
MSI198579	Chaum, D., "Smart Card 2000," 1991.
MSI198691	Global Projects Group, "Smart Card Technology International: The Global Journal of Advanced Card Technology," undated.
MSI199341	Vittal, J., "Active Message Processing: Messages as Messengers," 1980.
MSI199404	Chaum, D., "Achieving Electronic Privacy," August 1992.
MSI199417	Chaum, D., "Untraceable Electronic Cash," Extended Abstract, 1988.
MSI199425	Schaumüller-Bichl, S., "IC-Cards in High-Security Applications," undated.
MSI199480	Medvinsky, G., et al., "NetCash: A Design for Practical Electronic Currency on the Internet," 1993.
MSI199485	Dukach, S., "SNPP: A Simple Network Payment Protocol," December 1992.
MSI199525	Gifford, D., et al., "The Cirrus Banking Network," August 1985.
MSI199536	National Semiconductor "iPower Technology," undated.
MSI199550	Abadi, M., et al., "Authentication and Delegation with Smart-Cards," October 22, 1990, revised July 30, 1992.
MSI199576	Carnegie Mellon University, "Internet Billing Server," Prototype Scope Document, INI Tech Report, October 14, 1993.
MSI199605	Krajewski, Jr., M., et al., Concept for a Smart Card Kerberos," 1992.
MSI199612	Krajewski, Jr., M., "Smart Card Augmentation of Kerberos," February 1993.
MSI199618	Krahewsjum Jr., M., "Applicability of Smart Cards to Network User Authentication," 1994.
MSI199633	Harty, K., et al., "Case Study: The VISA Transaction Processing System," undated.

Exhibit F Publications

MSI199645	International Standard, "Bank Card Originated Messages: Interchange Message Specification Content for Financial Transactions," August 15, 1987.
MSI199681	Rivest, R., "The MD5 Message-Digest Algorithm," April 1992.
MSI199700	Voydock, V.L., et al., "Security Mechanisms in High-Level Network Protocols," June 1983.
MSI199736	Needham, R.M., "Ading Capability Access to Conventional File Servers," January 1979.
MSI199738	Gligor, V.D., et al., "Object Migration and Authentication," November 1979.
MSI199743	Chaum, D.L., et al., "Implementing Capability-Based Protection Using Encryption," July 17, 1978.
MSI199755	Gifford, D.K., "Cryptographic Sealing for Information Secrecy and Authentication," April 1982.
MSI204966	Johnson, H.L., et al., "A Secure Distributed Capability Based System," 1985.
MSI204983	Tanenbaum, A.S., et al., "Distributed Operating Systems," December 1985.
MSI205081	Voydock, V.L., et al., "Security Mechanisms in High-Level Network Protocols," June 1983.
MSI205118	Rushby, J.M., "Design and Verification of Secure Systems," 1981.
MSI205137	Clark, P.C., et al., "BITS: A Smartcard Protected Operating System," November 1994.
MSI205190	Brumm, P., et al., "80386/80486 Assembly Language Programming," 1993.
MSI205529	Hutt, A.E., et al., "Computer Security Handbook," 1988.
MSI205548	Brown, C.W., "Security for Minicomputers and Microcomputers," (undated).
MSI205621	The Risks Digest, "Forum on Risks to the Public in Computers and Related Systems," Vol. 15; Issue 39, January 21, 1994
MSI205633	The Risks Digest, "Forum on Risks to the Public in Computers and Related Systems," Vol. 15; Issue 47, February 9, 1994.
MSI205648	Halfhill, T.R., et al., "Agents on the Loose," February 1994.
MSI205650	Wayner, P., "Agents Away," May 1994.

Exhibit F Publications

MSI205659	Hawk, H.S., "RSA & General Magic," email to Good Guys, January 6, 1994.
MSI205660	Byte.com, "Speaking the Same Language," May 1994.
MSI205749	Atkins, D., et al., "The Magic Words are Squeamish Ossifrage," (undated).
MSI205967	Schill, A., et al., "Mobility Aware Multimedia X. 400 e-mail: A Sample Application Based on a Support Platform for Distributed Mobile Computing," undated.
MSI205981	National Institute of Standards and Technology, "History of Computer Security: Early Computer Security Papers, Part 1," 1998.
MSI206028	Department of Defense Standard, "Department of Defense Trusted Computer System Evaluation Criteria," December 1985.
MSI206144	Department of Defense Computer Security Center, "Department of Defense Password Management Guideline," April 12, 1985.
MSI206178	"Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments," http://www.radium.ncsc.mil/tpep/library/rainbow/CSC-STD-004-85.html, June 25, 1985.
MSI206672	Forcht, K.A., "Computer Security Management," 1994.
MSI206764	Olivier, M.S., et al., "DISCO: A Discretionary Security Model for Object-Oriented Databases," 1992.
MSI206777	Olivier, M.S., "A Multilevel Secure Federated Database," 1994.
MSI206951	"iOpener," Registered Trademark of National Semiconductor Corporation, Registration date October 4, 1994.
MSI206952	U.S. Patent and Trademark Coversheet for National Semiconductor Corporation "iOpener" Trademark, Registration date October 4, 1994.
MSI206982	UNKNOWN
MSI207089	Leary, P., "Are There Ciphers in Shakespeare?," 1995.
MSI207745	Toohey, J., "Using OLE 2.X in Application Development," 1994
MSI207781	Holzner, S., "Heavy Metal OLE 2.0 Programming," 1994.
MSI207820	Sheridan Software Systems, "Data Widgets 2.0: OLE Controls for Database Application Development," 1993-1995.

Exhibit F Publications

MSI207847	van Gilluwe, F., "The Undocumented PC: A Programmer's Guide to I/O, Cpus, and Fixed Memory Areas," 1994.
MSI207873	Short, K.L., "Microprocessors and Programmed Logic," 1981.
MSI207998	Table of Contents: Motorola MC68030 User's Manual, undated.
MSI208023	Mullender, S., "Distributed Systems," 1989.
MSI208222	Orfali, R., et al., "The Essential Distributed Objects Survival Guide," 1996.
MSI208280	Lockhart, Jr., H.W., "OSF DCE Guide to Developing Distributed Applications," 1994.
MSI208321	Open Software Foundation, "OSF DCE Administration Guide-Core Components," 1993.
MSI208364	Rosenberry, W., et al., "Distributing Applications Across DCE and Windows NT," 1993.
MSI208434	Shirley, J., "Guide to Writing DCE Applications," 1st Ed. 1992.
MSI208465	Shirley, J., et al., "Guide to Writing DCE Applications," 2nd Ed. 1994.
MSI208843	Honeyman, P., "Digest of the First UNSENIX Workshop on Electronic Commerce (EC 95)," July 1995.
MSI209487	Rubin, A.D., "Trusted Distribution of Software Over the Internet," 1995.
MSI209496	Tanenbaum, A.S., et al., "Experiences with the Amoeba Distributed Operating System," 1990.
MSI209525	Tanenbaum, A.S., et al., "The Amoeba Distributed Operating System-A Status Report," 1991.
MSI209547	Tanenbaum, A.S., et al., "The Amoeba Distributed Operating System," 1990.
MSI209692	Kim, G.H., et al., "The Design and Implementation of Tripwire: A File System Integrity Checker," November 19, 1993.
MSI209713	Farmer, D., "The COPS Security Checker System," July 10, 1992.
MSI209854	Kohl, J.T., et al., "The Evolution of the Kerberos Authentication System," 1991.
MSI209981	Lewontin, S., et al., "The DCE Web Project: Providing Authorization and Other Distributed Services to the World Wide Web," February 22, 2002.

Exhibit F Publications

MSI210379	Mann, C.C., "Homeland Insecurity," September 2002.
MSI210403	Boone, J.V.,et al., "The Start of Digital Revolution: SIGSALY Secure Digital Voice Communications in World War II," December 10, 2002.
MSI210421	Weadon, P.D., "The SIGSALY Story," December 10, 2002.
MSI210447	"iOpener," Registered Trademark of National Semiconductor Corporation, Registration date October 4, 1994.
MSI210496	Schill, A.B., et al., "DC++: Distributed Object-Oriented System Support on top of OSF DCE," 1993.
MSI210517	Schill, A.B., et al., "DCE-The OSF Distributed Computing Environment Client Server Model and Beyond," October 1993.
MSI210522	Rosenberry, W., et al., "Understanding DCE," 1992.
MSI210838	Karger, P.A., et al., "A VMM Security Kernel for the VAX Architecture," 1990.
MSI210856	Landwehr, C.E, et al., "A Taxonomy of Computer Program Security Flaws," September 1994.
MSI210900	Landwehr, C.E., "Formal Models for Computer Security," September 1981.
MSI210941	Young, W.D., "Verifiable Computer Security and Hardware: Issues," September 1991.
MSI210984	Cybenko, G, et al., "Cognitive Hacking: A Battle for the Mind," 2002.
MSI211023	National Security Agency, "Security Enhanced LINUX," December 10, 2002.
MSI211026	National Security Agency, "Korean War Commemoration," December 10, 2002.
MSI211071	National Security Agency, "Guide to the Secure Configuration and Administration of Microsoft Exchange 5.x®," June 20, 2002.
MSI211133	National Security Agency, "Microsoft Office 97 Executable Content Security Risks and Countermeasures," December 20, 1999.
MSI211180	Bartock, P.F., et al., "Guide to Securing Microsoft Windows NT Networks," September 18, 2001.
MSI211325	Bickel, R., et al., "Guide to Securing Microsoft Windows XP," October 30, 2002.

Exhibit F Publications

MSI211504	National Security Agency, "SIGSALY Secure Digital Voice Communications in World War II," December 10, 2002.
MSI211515	Distributed Computing Environment, "DCE Technology at Work," December 13, 2002.
MSI211582	Kent, S., "Part II: Certificate-Based Key Management," February 1993.
MSI213017	Saydjari, O.S., et al., "LOCK Trek: Navigating Unchartered Space," 1989.
MSI213026	Saydjari, O.S., et al., "LOCK Trek: Navigating Unchartered Space," 1989.
MSI213042	Karger, P.A., et al., "Multics Security Evaluation: Vulnerability Analysis," June 1974.
MSI213076	Aucsmith, D., et al., "Common Data Security Architecture," January 22, 1996.
MSI213088	Jaeger, T, et al., "Support for the File System Security Requirements of Computational E-Mail Systems," November 1994.
MSI213206	Tirkel, A.Z. et al., "Electronic Water Mark," (undated).
MSI213211	van Schyndel, R.G., et al., "A Digital Watermark," (undated).
MSI213214	Kurak, C., et al., "A Cautionary Note On Image Downgrading," 1992.

EXHIBIT G - Patents

MSI023624-44
MSI023734-69
MSI023803-9
MSI029229
MSI029230
MSI036205-19
MSI080838-58
MSI082687-99
MSI088149
MSI088161
MSI088513
MSI088524-49
MSI088550
MSI089473-81
MSI089482-514
MSI089539-47
MSI089548
MSI089604
MSI089686-93
MSI089700-05
MSI089706-15
MSI089776
MSI089806-13
MSI089814-28
MSI089842-48
MSI089849-63
MSI089864-71
MSI160834-1126
MSI162608-29
MSI162630-45
MSI162959-3013
MSI163080-116
MSI163117
MSI163128-168
MSI163192-214
MSI163254-337
MSI163419
MSI163432
MSI163463
MSI163487
MSI163500
MSI163664
MSI163763
MSI163791
MSI168646
MSI168647-659
MSI168660-754
MSI173420-838
MSI184171-191
MSI187425-477
MSI187550-625
MSI2012502-22
MSI204898-932
MSI204933-965
MSI205493-499
MSI205500-522
MSI205828-844;
MSI212472-89
MSI205845-864
MSI207029-048
MSI207049-088
MSI210261-83
MSI210284-96
MSI210330-57
MSI210358-78
MSI210579-600
MSI210601-16
MSI210652-69
MSI211661-951
MSI212012-19
MSI212020-51
MSI212201-36
MSI212237-45
MSI212246-53
MSI212254-300
MSI212344-53
MSI212354-61
MSI212362-74
MSI212375-79
MSI212380-417
MSI212418-22
MSI212423-32
MSI212433-48
MSI212449-63
MSI212464-471
MSI2124690-501
MSI212615-41
MSI212642-62
MSI212663-74
MSI212675-97
MSI212698-706
MSI212736-43
MSI212744-61
MSI212762-69
MSI212770-78
MSI212779-90
MSI212791-807
MSI212808-40
MSI212841-60
MSI212861-76
MSI212877-87
MSI212888-901
MSI212902-10
MSI212911-21
MSI212922-49
MSI212950-62
MSI212963-73
MSI212974-3016
MSI213066-75
MSI213097-107
MSI213108-40
MSI213141-78
MSI213179-91
MSI213192-205
MSI213221-29
MSI213230-39
MSI213240-48
MSI213249-57
MSI213258-61

Exhibit G - Microsoft's Patent Local Rule 4-2 Disclosure (Limited to "Mini-Markman" Claims)

BATES NO.	PATENT NO.	ISSUE DATE	INVENTOR
MSI023624	5,138,712	August 11, 1992	Corbin
MSI023734	**5,113,518**	**May 12, 1992**	**Durst, Jr. et al.**
MSI023803	**4,941,175**	**July 10, 1990**	**Enescu et al.**
MSI029229	5,940,504	August 17, 1999	Griswold
MSI029230	4,020,326	April 26, 1977	Coulthurst
MSI036205	4,937,863	June 26, 1990	Robert et al.
MSI080838	**4,827,508**	**May 2, 1989**	**Shear**
MSI082687	4,609,777	September 2, 1986	Cargile
MSI088149	5,966,440	October 12, 1999	Hair
MSI088161	5,191,573	March 2, 1993	Hair
MSI088513	4,866,769	September 12, 1989	Karp
MSI088524	**5,291,598**	**March 1, 1994**	**Grundy**
MSI088550	**5,014,234**	**May 7, 1991**	**Edwards, Jr.**
MSI089473	3,996,449	December 7, 1976	Attanasio et al.
MSI089482	**4,104,721**	**August 1, 1978**	**Markstein et al.**
MSI089539	**4,183,085**	**January 8, 1980**	**Roberts et al.**
MSI089548	4,246,638	January 20, 1981	Thomas
MSI089604	4,442,484	April 10, 1984	Childs, Jr et al.
MSI089686	4,471,216	September 11, 1984	Herve
MSI089700	4,523,271	June 11, 1985	Levien
MSI089706	4,525,599	June 25, 1985	Curran et al.
MSI089776	4,562,305	December 31, 1985	Gaffney, Jr.
MSI089806	4,598,288	July 1, 1986	Yarbrough et al.
MSI089814	4,599,489	July 8, 1986	Cargile
MSI089842	4,609,985	September 2, 1986	Dozier
MSI089849	**4,621,321**	**November 4, 1986**	**Boebert et al.**
MSI089864	4,621,334	November 4, 1986	Garcia
MSI160834	**5,603,031**	**February 11, 1997**	**White et al.**
MSI162608	**5,577,209**	**November 19, 1996**	**Boyle et al.**
MSI162630	**5,369,702**	**November 29, 1994**	**Shanton**
MSI162959	**5,206,951**	**April 27, 1993**	**Khoyi et al.**
MSI163080	**5,724,425**	**March 3, 1998**	**Chang et al.**
MSI163117	4,995,082	February 19, 1991	Schnorr
MSI163128	5,689,565	November 18, 1997	Spies et al.
MSI163192	5,689,566	November 18, 1997	Nguyen
MSI163254	**5,649,099**	**July 15, 1997**	**Thelmer et al.**
MSI163419	4,816,655	March 28, 1989	Musyck et al.
MSI163432	**5,892,899**	**April 6, 1999**	**Aucsmith et al.**
MSI163463	**5,390,297**	**February 14, 1995**	**Barber et al.**
MSI163487	5,956,408	September 21, 1999	Arnold
MSI163500	5,625,693	April 29, 1997	Rohatgi et al.
MSI163664	4,259,720	March 31, 1981	Campbell
MSI163763	4,321,672	March 23, 1982	Braun et al.
MSI163791	5,636,276	June 3, 1997	Brugger
MSI168646	5,956,408	September 21, 1999	Arnold (File History)
MSI168647	5,956,408	September 21, 1999	Arnold
MSI168660	5,956,408	September 21, 1999	Arnold (Application)
MSI173420	5,390,297	February 14, 1995	Barber et al.
MSI184171	5,365,587	November 15, 1994	Campbell et al.
MSI187425	5,373,440	December 13, 1994	Cohen et al.
MSI187550	5,557,798	September 17, 1996	Skeen et al.

Exhibit G - Microsoft's Patent Local Rule 4-2 Disclosure (Limited to "Mini-Markman" Claims)

MSI212502	**5,490,216**	**February 6, 1996**	**Richardson, III**
MSI204898	**5,715,314**	**February 3, 1998**	**Payne et al.**
MSI204933	**5,724,424**	**March 3, 1998**	**Gifford**
MSI205493	**5,432,851**	**July 11, 1995**	**Scheidt et al.**
MSI205500	**5,369,707**	**November 29, 1994**	**Follendore, III**
MSI205828	5,432,928	July 11, 1995	Sherman
MSI212472	5,432,928	July 11, 1995	Sherman
MSI205845	**5,301,326**	**April 5, 1994**	**Linnett et al.**
MSI207029	**5,319,735**	**June 7, 1994**	**Preuss et al.**
MSI207049	**5,450,490**	**September 12, 1995**	**Jensen et al.**
MSI210261	**5,164,988**	**November 17, 1992**	**Matyas et al.**
MSI210284	5,699,427	December 16, 1997	Chow et al.
MSI210330	**4,926,480**	**May 15, 1990**	**Chaum**
MSI210358	**4,529,870**	**July 16, 1985**	**Chaum**
MSI210579	**5,361,359**	**November 1, 1994**	**Tajalli et al.**
MSI210601	5,325,524	June 28, 1994	Black et al.
MSI210652	5,802,590	September 1, 1998	Draves
MSI211661	6,016,393	January 18, 2000	White et al.
MSI212012	5,199,074	March 30, 1993	Thor
MSI212020	5,367,621	November 22, 1994	Cohen et al.
MSI212201	**5,129,084**	**July 7, 1992**	**Kelly, Jr. et al.**
MSI212237	**5,150,407**	**September 22, 1992**	**Chan**
MSI212246	**5,199,066**	**March 30, 1993**	**Logan**
MSI212254	5,204,897	April 20, 1993	Wyman
MSI212344	5,263,157	November 16, 1993	Janis
MSI212254	5,263,165	November 16, 1993	Janis
MSI212362	**5,276,901**	**January 4, 1994**	**Howell et al.**
MSI212370	**5,283,830**	**February 1, 1994**	**Hinsley et al.**
MSI212375	**5,287,407**	**February 15, 1994**	**Holmes**
MSI212380	**5,335,346**	**August 2, 1994**	**Fabbio**
MSI212418	**5,337,357**	**August 9, 1994**	**Chou et al.**
MSI212423	**5,343,526**	**August 30, 1994**	**Lassers**
MSI212433	5,349,642	September 20, 1994	Kingdon
MSI212449	5,359,721	October 25, 1994	Kempf et al.
MSI212464	5,371,792	December 6, 1994	Asai et al.
MSI212490	5,440,634	August 8, 1995	Jones et al.
MSI212615	5,845,281	December 1, 1998	Benson et al.
MSI212642	**4,529,870**	**July 16, 1985**	**Chaum**
MSI212663	4,573,119	February 25, 1986	Westheimer et al.
MSI212675	**4,578,530**	**March 25, 1986**	**Zeidler**
MSI212698	**4,590,552**	**May 20, 1986**	**Guttag et al.**
MSI212736	**4,683,968**	**August 4, 1987**	**Appelbaum et al.**
MSI212744	4,780,821	October 25, 1988	Crossley
MSI212762	**4,796,220**	**January 3, 1989**	**Wolfe**
MSI212770	4,864,616	September 5, 1989	Pond et al.
MSI212779	4,868,736	September 19, 1989	Walker
MSI212791	4,888,798	December 19, 1989	Earnest
MSI212808	4,893,248	January 9, 1990	Pitts et al.
MSI212841	4,893,332	January 9, 1990	Brown
MSI212861	**4,953,209**	**August 28, 1990**	**Ryder, Sr. et al.**
MSI212877	4,962,533	October 9, 1990	Krueger et al.
MSI212888	4,975,878	December 4, 1990	Boddu et al.
MSI212902	5,022,080	June 4, 1991	Durst et al.
MSI212911	5,027,397	June 25, 1991	Double et al.

Exhibit G - Microsoft's Patent Local Rule 4-2 Disclosure (Limited to "Mini-Markman" Claims)

MSI212922	**5,032,979**	**July 16, 1991**	**Hecht et al.**
MSI212950	5,058,162	October 15, 1991	Santon et al.
MSI212963	5,065,429	November 12, 1991	Lang
MSI212974	**5,109,413**	**April 28, 1992**	**Comerford et al.**
MSI213066	5,383,113	January 17, 1995	Kight et al.
MSI213097	5,524,933	June 11, 1996	Kunt et al.
MSI213108	4,672,605	June 9, 1987	Hustig et al.
MSI213141	5,103,459	April 7, 1992	Gilhousen et al.
MSI213179	3,845,391	October 29, 1974	Crosby
MSI213192	DE 29 43 436 A1	October 26, 1979 (filed)	Szepanski
MSI213221	5,574,962	November 12, 1996	Fardeau et al.
MSI213230	5,581,800	December 3, 1996	Fardeau et al.
MSI213240	5,787,334	July 28, 1998	Fardeau et al.
MSI213249	5,721,788	February 24, 1998	Powell et al.
MSI213258	5,079,648	January 7, 1992	Maufe